Hierarchies for Constrained Partial Spanning Problems in Graphs

Miklos Molnar

I R I S A

INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTÈMES ALÉATOIRES

P U B L I C A T I O N
I N T E R N E
N° 1900

HIERARCHIES FOR CONSTRAINED PARTIAL SPANNING PROBLEMS IN GRAPHS

MIKLÓS MOLNÁR

ISSN 1166-8687

IRISA
CAMPUS UNIVERSITAIRE DE BEAULIEU - 35042 RENNES CEDEX - FRANCE

Institut de Recherche en Informatique et Systèmes Aléatoires
Campus de Beaulieu – 35042 Rennes Cedex – France
Tél. : (33) 02 99 84 71 00 – Fax : (33) 02 99 84 71 71
http://www.irisa.fr

Hierarchies for Constrained Partial Spanning Problems in Graphs

Miklós Molnár *

Systèmes communicants
Projet At-Ren

Publication interne n˚ 1900 — July 2008 — 47 pages

Abstract: The problem of minimal cost partial spanning trees is well known as the NP-complete Steiner problem in graphs. If there are constraints to respect by using some edges and nodes in the spanning structure or constraints on the required end to end (QoS) properties of the partial spanning structure, then the optimal structure can be different from a partial spanning tree. This paper present simple hierarchical spanning structures which correspond generally to the optimal solution. To illustrate the optimality of spanning hierarchies, two problems in communication networks, which are very usefull for multicast routing, are analyzed. In the first problem, the constraints are related to the physical capacity of the nodes in WDM networks. Namely, the analyzed example presents the case of sparse splitter nodes in optical networks. In the second example, the multi-constrained multicast routing is analyzed. In both cases, the optimal routing structure is a hierarchy. Since its computation is NP-complete, after the presentation of the computation difficulties some ideas on heuristic algorithms are discussed.

Key-words: Graph theory, spanning problems, partial spanning trees, optimization with constraints, hierarchies, minimal cost hierarchies, networks, multicast routing, optical networks, multiconstrained multicast routing

(Résumé : tsvp)

* miklos.molnar@irisa.fr

Centre National de la Recherche Scientifique
(UMR 6074) Université de Rennes 1 – Insa de Rennes

Institut National de Recherche en Informatique
et en Automatique – unité de recherche de Rennes

Hiérachies pour les problèmes de recouvrement partiel sous contraintes dans les graphes

Résumé : Le problème NP-difficile de l'arbre couvrant partiel de longueur minimal est connu sous le nom du problème de Steiner. Quand il y a des contraintes dans le graphe pour utiliser certains sommets et/ou arêtes ou bien des critères pour la qualité de service des routes multicast sont donnés, la structure optimale n'est plus un arbre mais une hiérarchie. Ce rapport introduit la structure hiérarchique comme la généralisation de l'arbres et présente l'optimalité des hiérarchies dans certains cas. Pour illustrer cette optimalité, deux cas importants sont présentés. Dans le premier problème, le routage multicast dans les réseaux optiques WDM est analysé. Ici, les contraintes sont imposées sur l'utilisation des commutateurs et des liens optiques. L'exemple présente le cas où tous les commutateurs ne peuvent pas dupliquer la lumière. Dans le deuxième exemple, le routage multicast multicritère est analysé. Dans les deux cas, la route optimale correspond à une hiérarchie. Les problèmes d'optimisation présentés sont NP-difficiles. Quelques algorithmes exactes pour construire des hiérarchies optimales ainsi que des idées sur les algorithmes inexactes sont discutés.

Mots clés : Théorie de graphe, structures couvrantes, arbres couvrants partiels, optimisation sous contraintes, hiérarchies, hiérarchies de coût minimal, réseaux, routage multicast, réseaux optiques, routage multicast multicritère

1 Introduction

One of the first studies related to graphs corresponds to the computation of walks. The famous problem of Königsberg Bridges discussed by Euler in 1736 examined the possibility of a particular walk in a given graph: a walk passing only once by each edge. Moreover a lot of applications need the computation of shortest paths between node pairs in a given graph. Shortest paths are also special walks in graphs. Often, walks and spanning problems are related. A path can be considered as a structure spanning the node pair at the end points. So, the shortest path corresponds to the partial minimal spanning structure of the node pair. Minimal length Eulerian and Hamiltonian cycles (if they exist) spanning the edges or the nodes of a given graph correspond also to special walks. When a set of nodes containing more than 2 nodes should be spanned, the spanning sub graph with minimal length is a tree. Generally, trees are not considered as walk structures, but - as we shall see next - multiplying "walking individuals" at some nodes (*i.e.* at the branching nodes of the tree) permits to describe a hierarchical walk in the graph. This paper aims at special partial spanning problems. These optimal spanning problems of node sets are proposed and discussed in recent communication networks.

To position our problems, we propose a rapid survey of the most important related concepts. The definitions can be found for example in [1] [2]. A *walk* is given by a sequence of vertices and edges forming a connected sub graph beginning and ending with a vertex:

$$W = (x_0, e_0, x_1, e_1, x_2, e_2, x_3)$$

A walk can be *closed*: in this case its first and last vertices are the same, and it can be *open*: if they are different. Open walks are called *paths* or *chains* using directed or not directed concept respectively. In the following, we use only the directed path concept. Closed walks are called *cycles*. Often the concept of *trail* is used to describe walks in which no edge is repeated. A walk is *elementary* (or simple), if no vertices (and thus no edges) are repeated in the walk. Numbers of times the path and cycle concepts are used for elementary paths and cycles. To reference not elementary paths in the paper, we propose to use the term *itinerary*. A good symbolic model can be obtained using *tokens* to describe walking problems and walks. A walk can be described with the help of the propagation of a walking unit (or token). For example, let be $W = (x_0, e_0, x_1, e_1, x_2, e_2, x_0)$ a closed walk. A token starting from the vertex x_0 passes by the vertices x_1 and x_2 to return to x_0 using the edges e_0, e_1 and e_2. In traditional walking, only one token is used and it is sufficient to walk a path or a cycle.

The concept of *tree* is used to describe connected, elementary graphs without cycles. Generally, trees are not considered as walks. Let be a tree given as follows:

$$T = (x_0[e_0, x_1[e_1, x_2][e_2, x_3]])$$

The analogy with path description is trivial. The difference between paths and trees is that in trees a node can have several successors. In us eye, trees can be considered as elementary open walks, in which the walk can simultaneously be continued toward several adjacent nodes from some branching nodes. In other word, the token realizing the walk should be multiplied in the branching nodes of the tree. In our example, a token starting from the vertex $x - 0$ passes to x_1 using the edge e_0. In x_1 the token is duplicated; one of the tokens goes to x_2 using the edge e_1 and the other goes to x_3 using the edge e_2. Let us notice that a token can be duplicated at a node, but two token can never be unified by walking on a tree.

To handle some special group communication problems, we propose the generalization of the walk concept to trees and more generally to hierarchical walks in graphs.

In *hierarchical walks* the tokens realizing the walk are also duplicated in some vertices. After the duplication, the tokens continue the walk independently and there is no relation between the duplicated tokens. So, to speak about hierarchical walks, there is not possible to synchronize the separated tokens. A counter-example can facilitate the comprehension of this concept. Many times, token based descriptions of systems are given by using the well-known concept of Petri nets ([3]). In Petri nets, tokens can be duplicated in some vertices and they can synchronized (unified) elsewhere. Petri nets in which there are transitions synchronyzing two or more tokens can not be considered as hierarchical walks of tokens. From our point of view, the synchronization of separated tokens closes the "walk" and implicates cycles.

An important part of walk computation problems considers spanning problems. Generally, in spanning problems a set of vertices is given and must be spanned by a connected sub graph. Often, the spanning sub graph is a walk. If there is two vertices to span, the objective is to find a path between them and numbers of times to find a shortest path. If more than two vertices should be covered by the connected sub graph then generally a spanning tree is searched. If the spanning tree should be a tree with minimal length and a sub set of vertices is given, then the solution corresponds to a Partial Minimum Spanning Tree or Steiner tree. The problem of finding a Steiner tree is NP-hard and several propositions for exhaustive and approximated solutions exist (cf. a large state of the art for example in [4]).

Several problems can not be described with elementary walks. For example, random walks do not correspond to elementary walks. In this paper, we will demonstrate that some optimal spanning problems of vertex sets under constraints can not be resolved using elementary tree structures. The optimal solution of partial spanning problems in graph corresponds always to a hierarchy (and spanning trees are special cases of spanning hierarchies).

Hierarchies are often present in the graphs. An interesting related application domain is the exploration of the Web. The structure of the Web pages with the existing links between them corresponds to a highly hierarchical graph containing also cycles. The behavior of a user browsing on the Web can be considered as a (in some cases determined, in other cases random) walk in this graph. An interesting two layer graph model with random exploration strategy can be found in [5]. Let us notice that a human operator realizes generally a sequential walk to explore documents but sometimes the parallel look at several documents can also be imagined. Using automatons, the duplication of automatons is possible, so the exploration can be made in a hierarchical way. Some transportation problems with fixed and sparse entrepôts can also be considered as hierarchical walks in the transport graph. In our work, we are interested with data forwarding problems in communication networks. More precisely, we propose the study of non elementary hierarchical structures for multicast communications with different constraints. In this paper we deal with two kinds of multicast routing problems. In the first one, constraints are given on the capability of networks elements. After the related definitions in Section 2, Section 3 presents this routing problem in all optical networks. In the second problem the constraints are established on the end-to-end quality of service parameters of the multicast communication. The analysis can be found in Section 4. The analysis is followed by some perspectives of this work.

2 Hierarchies in graphs

The most frequently used concepts in graphs are related to walks. Well known walk structures are chains, paths, circuits, etc. Generally, in practical cases, the used walks are elementary walks: no edge (and so no vertex) of the graph is repeated in the walk. Contrarily, non elementary paths and cycles can describe arbitrary walks with repetitions of some edges and/or nodes.

Hierarchical objects are manipulated for a long time and in several domains. In the following, we limit our analysis on hierarchies in graphs. Generally, a hierarchy contains a parent node and some eventual children at each level of the structure. Traditional hierarchy structures in graphs corresponds to trees:

$$T = (v_0(e_1, v_1, e_2, (v_2(e_3, v_3, e_4, v_4)), e_5, v_5)) \tag{1}$$

In T each node is a unique node. At each level of a non empty tree, there is a root node connected to eventual sub-trees. So, a tree can be given recursively:

$$T = (v(e_1, T_1, e_2, T_2, \ldots, e_k, T_k)) \tag{2}$$

Using the concept of tokens, trees correspond to elementary walks in graphs where the tokens can be duplicated in some nodes but after the duplication two token can not be merged. With the help of a simple generalization, non elementary tree-like walks with duplication possibilities can be given. Similarly to non elementary paths, the repetition of some nodes in the tree-like structures can offer a more general structure. The following definition gives the extension of the tree concept to hierarchical structures in graphs.

Definition 1 (Hierarchy) *A hierarchy in a graph is given by a root node and by an eventual set of sub hierarchies connected to the root with an arc/edge of the graph. An empty hierarchy does not contain root nor sub hierarchies.*

A hierarchy can be considered as a two dimensionally ordered enumeration of nodes and edges which is not obligatory exempt from repetitions. In term of tokens, a hierarchy corresponds to a walk with eventual duplication of tokens in the nodes but without fusion of the tokens. A simple example of a hierarchy can be the following enumeration of nodes and edges:

$$H = (v_0(e_1, (v_1(e_3, v_3, e_4, (v_4(e_5, v_3)))), e_2, (v_5(e_6, v_6, e_7, v_4)))) \tag{3}$$

The same hierarchy can be described recursively:

$$H = (v_0(e_1, H_1, e_2, H - 2)) \tag{4}$$

where H_1 and H_2 are the sub-hierarchies of the root node v_0.

Figure 1 illustrates a hierarchy in a undirected graph. The nodes of the graph participate on different levels of the hierarchy shown in Figure 1/b). The projection of the hierarchy on the original graph is also presented in Figure 1/a).

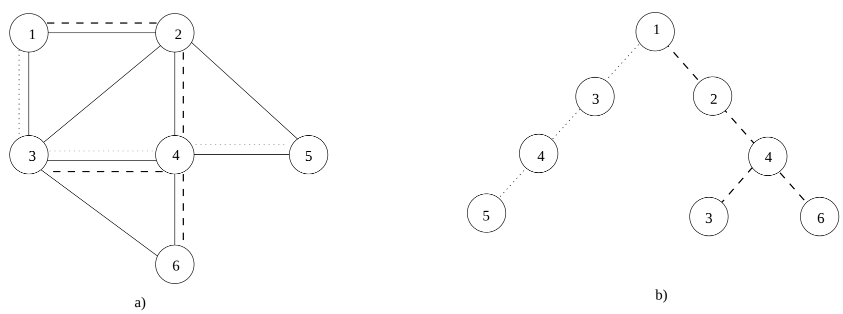

Figure 1: Example of a hierarchy in a undirected graph

Let us notice that hierarchies are more flexible than trees as it is expressed by the following properties.

Property 1 *A hierarchy can use the same node several times.*

Property 2 *An edge/arc of the referred graph can be used by a hierarchy several times.*

Property 3 *A not empty hierarchy is connected. (This property is given by the construction of a hierarchy: i.e.. the root of an arbitrary level is connected to the eventual sub hierarchies.)*

Property 4 *In a hierarchy, each node occurrence has only one predecessor. So, a hierarchy does not contain loops in the hierarchical enumeration. (In other words, there is no fusion of tokens. This property follows also from the construction of the hierarchy.)*

Let us notice that the projection of a hierarchy can contain loops in the related graph but the outstretched hierarchy itself does not contain loops. In the following we use the term *hierarchy* to describe the defined outstretched tree-like structure of the nodes independently from the repetitions and loops (cf. in Figure 1/b) and we use the term *projection of the hierarchy* when describe the same sequences of nodes in the original graph (cf. in Figure 1/a).

Hierarchies can be directed or not. If the hierarchy is defined in an oriented graph, then it is directed of course (and the arcs of the hierarchy should be directed in the same direction as the corresponding arcs in the graph in question). For potential application reason, the following two cases of homogeneously directed hierarchies are more important than the heterogeneously directed hierarchies.

Definition 2 (Upward hierarchy) *In an upward hierarchy, at each level the arcs are directed to the root node.*

Definition 3 (Downward hierarchy) *In a downward hierarchy, all the arcs are directed from the root to the sub hierarchies.*

In order to describe runs in the hierarchies and to distinguish the possible runs from the classic notion of paths in the related graph, we propose the definition of the following concept in the hierarchies.

Definition 4 (Itinerary) *An itinerary between two nodes n_1 and n_2 corresponds to the path between the given nodes in the hierarchy.*

An itinerary can be directed or not. A given itinerary is always a shortest path in the hierarchy but can be an arbitrary path (cycle, etc.) in the projection (in the related graph). An example of itinerary can be seen in Fig. 2. Often, we will talk about itineraries between nodes and the root of the hierarchy.

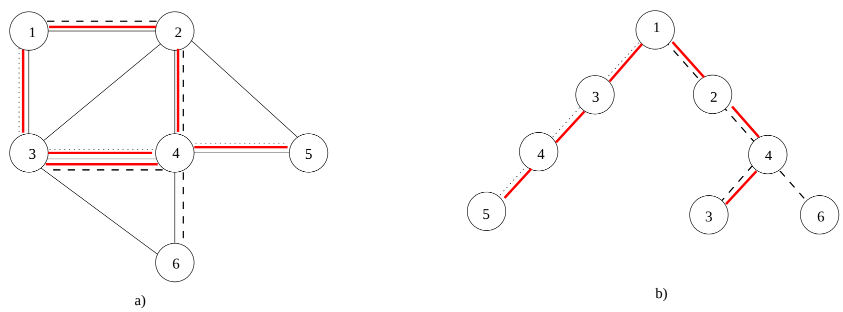

Figure 2: Example of an itinerary in the hierarchy given by Fig. 1

Due to Property 3 hierarchies are connected. Trivially the following properties are also trues.

Property 5 *In a undirected hierarchy there is one and only one itinerary between any node pair.*

Property 6 *In an upward (respectively downward) hierarchy there is one and only one directed itinerary between the root node and another node.*

Particular cases of hierarchies can be defined using constraints. In our recent study, the following constraints are introduced.

Constraint 1 (Constraint on the edges) *This constraint is imposed when in an undirected hierarchy related to an undirected graph an edge of the graph can belong to the hierarchy at most once.*

Constraint 2 (Constraint on the edges for directed usage) *In this case, in a directed hierarchy related to an undirected graph, an edge of the graph can be used in the same direction by the hierarchy at most once.*

Constraint 3 (Constraint on the arcs) *In a directed hierarchy related to a directed or undirected graph, an edge/arc of the related graph can belong to the hierarchy only once.*

Constraint 4 (Constraint on the nodes) *Applying this constraint, each node of the related graph can be used in the hierarchy only once.*

Constraint 5 (k-constraints) *The previously mentioned constraints can be generalized to express that each node and/or edge can be used in the hierarchy at most k times.*

Property 7 *Constraint 4 on the nodes is more strict than Constraints 1 2 and 3. Trivially, if not a single node can belong two times to the hierarchy then none of the edges/arcs can belong two times to the hierarchy.*

Important occurrences of particular hierarchies under constraints are well known. For example, trees (and so simple paths) are hierarchies under Constraint 4 on the nodes (cf. also lemma 2). Multi-trees (cf. [6]) can be considered as hierarchies under Constraint 2 on the edges for directed constructions. Namely, in a multi-tree, some edges of the related undirected graph are used in the two opposite directions to form two directed trees. Figure 3 illustrates such a multi-tree spanning the nodes of a given graph. This hierarchy can be applied as a good fault tolerant diffusion structure for multicast communications in optical networks.

Because repetitions are possibles, the number of nodes of a hierarchy without constraint can be limitless. For example, an infinite loop can be represented as an infinite path which corresponds to an infinite hierarchy. For practical purposes, in the following we focus only on finite hierarchies. Hierarchies can be used to resolve spanning problems. They can cover the whole set of the nodes in the related graph or a subset of the nodes. In this latter case we propose to talk about *partial spanning hierarchies.*

Hierarchies have trivial but important properties. The following lemma expresses the most important property of hierarchical structures.

Lemma 1 *An outstretched hierarchy (and not its projection) is exempt from cycles.*

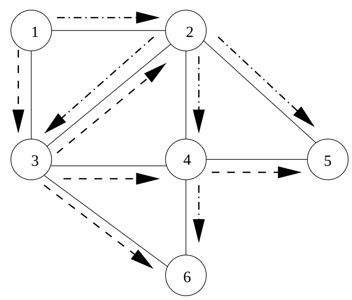

Figure 3: A multi-tree is a hierarchy under constraint on the edges for directed usage of the edges

Proof. By definition a hierarchy is always connected (cf.Property 3). Let each node occurrence be differentiated in the hierarchy. There is only one itinerary between two arbitrary node occurrences in the hierarchy (and so, there is no cycles). This property is trivially true in the hierarchies which contain only leaves and a root node. If the property is true in the sub-hierarchies of a hierarchy H_v rooted in a node v, it is also true in H_v by induction and applying the definition 4. ■

Lemma 2 *A tree is a hierarchy.*

Proof. The proof is trivial. It is sufficient to establish Constraint 4 to exclude multiple occurrences of a node in the hierarchy. The hierarchies under this constraint correspond to the definition 1 and are trees. ■

Often in spanning problems a subset $M \subset V$ of nodes should be covered and the better (*i.e.* the minimal cost) spanning structure is wanted. With the help of the hierarchies, the well known Steiner problem can be generalized as follows.

Definition 5 (Partial minimum spanning hierarchy) *A partial minimum spanning hierarchy of M in a graph $G = (V, E)$ is the hierarchy with the minimal cost from the set of partial spanning hierarchies covering the given group M.*

To facilitate the presentation of the partial spanning hierarchies, we propose the following definition of the more interesting nodes in the spanning problems.

Definition 6 (Significant node) *We will call the source, the destination nodes and the branching nodes (with a degree ≥ 2) as the significant nodes in the hierarchy.*

The following, trivial lemma can be interpreted both in directed and undirected graphs.

Lemma 3 *A partial minimum spanning hierarchy for a given group $M \subset V$ without any constraint in the graph $G = (V, E)$ corresponds to the partial minimum spanning tree of M.*

Proof. If there is no constraint for the hierarchy, then the partial minimum spanning hierarchy does not contain a same edge/arc nor a same node two times. Let us suppose that the optimal hierarchy H contains a node v twice. Let v' and v'' be the two occurrences. The sub-hierarchies of v'' in H can be connected arbitrarily to v' or to v''. Let these sub-hierarchies be connected to v'. So v'' becomes a leaf and it is useless to span M because the node v is already covered even if $v \in M$ (cf. Figure 4). In this case, v'' and the path from the last significant node to v'' can be deleted and a new shortest hierarchy can be obtained. This is in contradiction with the fact that H has minimal length. So the optimal hierarchy is without cycle and corresponds to a tree. Among the trees spanning a given node group, there is at most a minimal cost tree named partial minimum spanning tree (or Steiner tree, cf. a study on the Steiner trees in [4]). ■

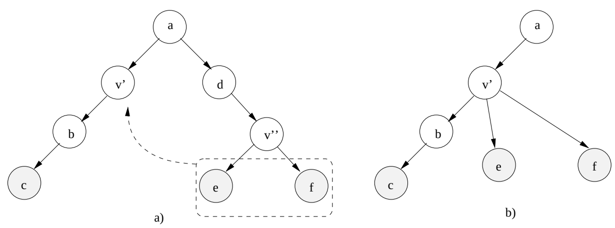

Figure 4: A node is present twice in a hierarchy without constraint

Interesting optimization problems can be formulated (especially for multicast communications) with the help of the hierarchies. As we shall see next, in some problems the spanning structure with minimal length does not correspond to a tree but to a hierarchy. In the next two sections, we present two different problems to illustrate the interest of hierarchies. Both in these cases they are special minimum spanning hierarchies which correspond to optimal solutions.

3 Partial minimum spanning hierarchies under constraint on the node degrees

In the first example, we propose to examine optimal routing structures for multicast communications under constraints in all optical WDM networks. The technological background is very well resumed in [7], where physical constraints are presented as well as some routing algorithms for multicast communications (cf. Chapter 7 in the book). In WDM optical networks the communications use continuous lightpaths from the source to the destination node. For source based multicast communications "light-trees" are proposed. We shall see next that optimal multicast routing structures do not always correspond to trees, but a first time we propose to talk about light-trees to facilitate the problem description. There are three important constraints to build multicast "trees" in WDM networks.

Constraint 6 (Wavelength continuity) *In simple optical networks where the wavelength conversion is not possible in the nodes, a "light-tree" should use the same wavelength from the source to the destinations.* [1]

Constraint 7 (Wavelength unicity constraint in the fibers) *In a given fiber between two optical switches two different lightpaths or light-trees or two segments of a same lightpath or light-tree can not use the same wavelength (because of the eventual interferences). This constraint for optical routing application is equivalent to the earlier given Constraint 2.* [2]

In the branching nodes of the multicast trees (nodes with degree more than 2) to realize the splitting of the incoming light toward the multiple outgoing fibers special optical instruments are needed. These instruments called splitter are expensive and are not availables in all of the network nodes. Generally, let us suppose that the splitting capability of a node is given by an integer value $d \geq 1$ given the maximal number of the outgoing signals for an incoming one. A value $d = 1$ indicates a splitting incapable switch.

Constraint 8 (Constraint on the light splitting) *The degree of the nodes in a light-tree can not exceed the splitting capability of the corresponding optical switch.*

So, in optimal cases, branching nodes of multicast "trees" coincide with splitting capable nodes of the network.

Routing structures used for multicast in WDM networks must satisfy the three constraints. To find good structures, we propose to study the optimal solution.

3.1 Problem formulation

Let us suppose that the links can be used both in the two directions and the topology of the optical network is given by an undirected graph $G = (V, E)$. The multicast group is given by a source node $s \in V$ and a set $D \subset V$ of destination nodes. As multicast route, a directed sub graph spanning the source and the set of the destinations is needed. Let $MC \subset V$ be the set of splitting capable nodes (accordingly, the nodes in $MI = V \setminus MC$ can not duplicate the light). Corresponding to Constraint 8 only the nodes in MC can have a degree greater than two in the multicast routing structure. Constraint 7 implicates that two arcs of the hierarchy can not use the same edge of the topology graph in the same direction (this constraint correspond to Constraint 2). The optimal solution having the minimal length between the sub graphs spanning $\{s\} \cup D$ is wanted.

Problem 1 *Minimum directed spanning sub graph with constraint on the edges and on the nodes.*

---

[1] In certain optical networks some switches contain elements to convert the wavelength of some lightpaths to another wavelength. In our study we basically suppose that there is no wavelength converter in the switches.

[2] In certain bidirectional fibers, the same wavelength can be used in both directions. Our hierarchy constructions can be easily adjusted to the use of these fibers.

Let $G = (V, E)$ be a undirected graph valuated with positive values on the edges, $s \in V$ a source node and $D \subset V$ a set of destinations. The problem corresponds to finding the minimum length directed graph G_M^* spanning $\{s\} \cup D$, having at least a directed path from the source to each destination with respect the degree constraint on the nodes and the constraint on the edges.

Lemma 1 *The optimal solution G_M^* of Problem 1 is a directed hierarchy (which does not always correspond to a tree). In other word: in the minimal length solution, there is no two different itineraries from the source to the same destination.*

Proof. Trivially, G_M* is a connected component covering the members of the group. In this connected and directed graph, a directed itinerary from the source s to each destination must be existed. Let us suppose that this structure does not correspond to a hierarchy. In this case, at least one node v exists which has an inferior degree $k > 1$ in the structure (cf. Lemma 1). By deleting $k-1$ incoming arcs of v, the connectivity of the nodes to the source is preserved and a shortest spanning structure is obtained. So, the supposed not hierarchical structure can not be the optimal solution.

The optimal structure does not always correspond to a tree in the graph: *i.e.,* a same node of the graph can belong several times to the hierarchy. To prove this, let us consider the example of Figure 5. In this example the source node is the node 1 and the destinations are the nodes 5 and 6. In the given graph only the node 2 can duplicate incoming messages and can correspond to a branching node. The minimum spanning structure is a hierarchy which uses the node 4 two times. ■

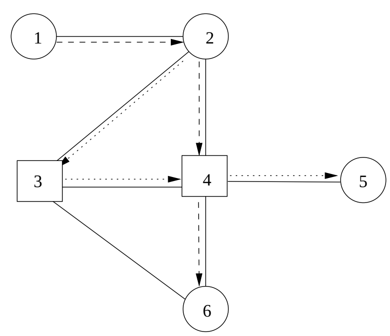

Figure 5: The optimal directed spanning structure is not a tree

A not hierarchical spanning sub graph is illustrated in Figure 6. In the given structure, the arcs $(2, 3)$, $(5, 6)$ and $(5, 4)$ can be deleted without loss of the connectivity from the source 1 to the destinations 3, 4 and 6.

Definition 7 (DMPSH under edge and degree constraints) *We propose to call the optimal solution G_M^* of Problem 1 as directed minimum partial spanning hierarchy (DMPSH) under edge constraint and degree constraint on the nodes.*

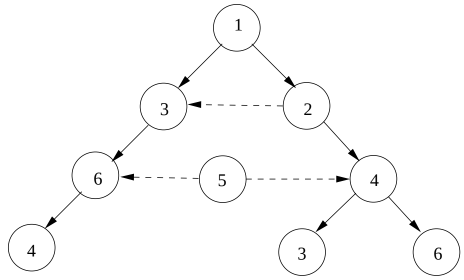

Figure 6: A set of itineraries forming cycles

Lemma 2 *At an arbitrary level of a DMPSH under edge constraint and degree constraint on the nodes, each sub hierarchy of the root is also a DMPSH under the same constraints.*

Proof. Let r be the root of a (sub) hierarchy of a DMPSH H under the edge constraint and the degree constraint on the nodes and let H_i^r be a sub hierarchy related to r. H_i^r should satisfy the two constraints otherwise H can not be a hierarchy satisfying them. Let r_i be the root of H_i^r and let D_i be subset of destinations spanned by H_i^r. H_i^r should be the hierarchy with minimal length between hierarchies spanning the set $\{r_i\} \cup D_i$ and satisfying the edge constraint and the degree constraint on the nodes. Let us suppose that it is not the minimum hierarchy and another hierarchy $H_i^{r'}$ under the same constraints and spanning the same node set exists. In this case H_i^r can be replaced by $H_i^{r'}$ in H resulting a hierarchy H'. The constraints are satisfied by H' and H' is shortest than H which is in contradiction with the definition of H. ■

Let us notice that the graph connecting the children r_i to r form a star corresponding to a PMST (Partial Minimum Spanning Tree) of $\{r\} \cup_i \{r_i\}$.

Lemma 3 *In a DMPSH under edge constraint and degree constraint on the nodes, a node $n \in MC$ with unlimited duplication possibility belong at most once to the hierarchy.*

Proof. Let us suppose that the node n is present two times in the DMPSH H which corresponds to edge constraint and degree constraint on the nodes: in the itineraries I_1 and I_2 (cf. Figure 7/a). Without loss generality, let us suppose that each occurrence of n has only one child in H. Let H_1 and H_2 the directed sub hierarchies of the two occurrences of n. Because n can duplicate incoming messages, one of H_1 and H_2 can be connected as a second child to the other occurrence of n and the occurrence of n without child can be deleted. The remaining graph corresponds to a directed hierarchy spanning the given nodes and satisfying the two conditions and it is shortest than H. The contradiction is trivial. So n can not be present twice in the minimal length directed hierarchy. ■

A particular case corresponds to the case where the node n is present twice on a same itinerary I (cf. Figure 7/b). In this case, the two incoming directed paths P_1 and P_2 are in

I. Similarly to the general case, one of them can be deleted taking into account the given constraints and without loss of the spanning property of the hierarchy. Namely, the second directed incoming path of n (the path P_2 in the figure) can be deleted and the obtained hierarchy is shortest.

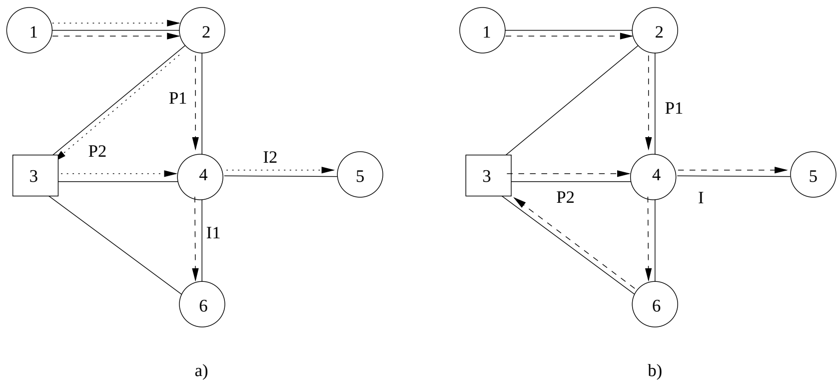

Figure 7: A node without constraint of degree is present only once in the hierarchy

Lemma 4 *In a DMPSH under constraints on the edges and degrees of the nodes, a node* $n \in MI$ *the degree of which is limited to 2 belong at most* $d = |D|$ *times to the hierarchy.*

Proof. At first we show that a node $n \in MI$ can not be repeated twice on a directed itinerary I_i from the source to a given destination d_i without a significant node between the two occurrences. We proof this with contradiction. Let us suppose that the node n is present two times in the directed itinerary I_i and there is no significant node between the two occurrences. In this case, this itinerary contains a directed cycle which can be eliminated to obtain a directed hierarchy spanning the same destinations without violation of the constraints. So the node n can be present only once on a segment of an itinerary which does not contain any significant node. Let us suppose that the node n is present several times in a DMPSH under constraints on the edges and degrees of the nodes. It is possible, if and only if the occurrences of n are on different itineraries or there is at most one significant node between two occurrences of n on a same itinerary. Since a significant node corresponds to a destination or to a branching node which leaves to another destination, each occurrence of n can be associated to a different destination. In the worst case, the same node n belong d times to the optimal hierarchy. ■ Figure 5 illustrates also the multiple presence of a node with degree constraint in the optimal hierarchy.

3.1.1 Optimal solution using the branch and bound technique

To limit the search space of the branch and bound algorithm, Constraints 2, 8 and in addition the results of Lemmas 3 and 4 can be applied. Namely:

- an arc (i, j) can not belong twice to the solution: in an arbitrary node i, only adjacent arcs which are not yet used in the hierarchy can be considered
- a MI node can have only one successor and can belong at most $d = |D|$ times to the result
- if a MI node belongs twice to an itinerary, then a significant node should be between the two occurrences in the itinerary (cf. the proof of Lemma 4)
- a MC nodes and the source belong only once to the result.

To search the DMPSH and built step by step the search tree with the branch and bound algorithm, we should define which kind of hierarchies can be considered as the successor hierarchies of a given hierarchy H.

Definition 8 *Let $G = (V, E)$ be an undirected graph and H a directed hierarchy already constructed in G. Let L be the set of leaves in H. A hierarchy H_s is a successor of H in G for the optimal solution search algorithm if $H \in H_s$ and $H_s \setminus H$ contains only arcs using adjacent edges of L (directed from the leaf of H).*

The construction of the hierarchy starts at the source node. The nodes of the search tree correspond to partial spanning hierarchies directed from the source to the other nodes. In each step of the branch and bound algorithm, the leaf of the search tree having the minimal cost/length is selected. If the hierarchy H_i corresponding to this node spans all of the destinations, then a minimal cost partial spanning hierarchy (the optimal solution) is found. Otherwise, the successor hierarchies $H_i^j, j = 1, ...k$ of this hierarchy should be computed and verified according to the constraints here before. Then the valid successors with their length are added as leaves to the search tree. We consider a hierarchy H_i^j as a successor of H_i if and only if new arcs of H_i^j are added using adjacent edges of the leaves of H_i (cf. definition 8). If the current hierarchy has not valid successors, its length becomes infinite.

In the following description of the algorithm, to each node is associated a partial spanning hierarchy and its length. Remember that the branch and bound algorithm examines only the set L of leaves of the search tree.

The branch and bound algorithm rakes the search tree from the root node which contains only the source node until the coverage of the multicast group.

Theorem 1 *The branch and bound algorithm finds the optimal solution.*

Proof. The optimal solution satisfies the constraints

Let H_i be a hierarchy corresponding to the given constraints and H_{i-1} be its predecessor in the search tree. Trivially $H_{i-1} \subset H_i$ and H_{i-1} satisfies the constraints. In this manner, there is a path from the source to the optimal solution in the search tree and this path contains only nodes satisfying the given constraints. The algorithm visits the leaves of the

Algorithm 1 Branch and bound algorithm to find DMPSH with constraints on the nodes and on the edges

Input: the graph $G = (V, E)$, the set MC of splitting capable nodes, the source s, the destination set D
Output: H the DMPSH under the constraints

```
H_0 ← s        {first hierarchy initialized with the source}
d(H_0) ← 0
K ← {(H_0, d(H_0))}        {set of the leaves in the search tree}
compute S the set of successor hierarchies of H_0
E_s ← adjacent_edges_of(s)
G' ← G \ {E_s}        {it is not possible to return to the source}
while K ≠ ∅ do
   select H_min the leaf in K with minimal length
   if M ⊂ H_m in then
      return H_min        {it is the optimal solution}
      STOP
   end if
   compute S the successor hierarchies of H_min using G'
   if S ≠ ∅ then
      for all H_s ∈ S do
         if H_s corresponds to Constraints 2, 8 and Lemmas 3,4 then
            K ← L ∪ {(H_s, d(H_s))}
         end if
      end for
   end if
   K ← K \ {(H_m in, d(H_m in))}        {delete H_min}
end while
```

search tree in an increasing order of the lengths and can not be stopped before visiting the solution (there is not an other solution with less length). ■

Usually, in tree based routing algorithm, if a single tree is not capable to span the destinations in a WDM network, then a set of trees (called light forest) rooted at the source node is applied for routing (cf. [8]). This solution is feasible if there are several available wavelengths in the network because the source node is capable to send the same messages using different wavelengths.

The same idea can be applied in the case of the hierarchies: in some case, a set of hierarchies may be built in order to diminish the overall cost of the routing structure. The determination of the optimal set of hierarchies (which has minimal length) is a little bit more complex as our problem and is not discussed here.

The complexity of the algorithm

In the worst case, the optimal solution is done by visiting the last hierarchy given by the branch and bound algorithm. So, the worst case complexity is equal to the maximal number of hierarchies visited by the algorithm. Remember that the number of nodes in G is $n = |V|$ and the number of splitting capable nodes is $p = |MC|$ and there are $d = |D|$ destinations in the group. Each hierarchy can contain at most once a splitting capable node and at most d times the other nodes. Figure 9 presents the search tree after the first three steps of the branch and bound algorithm on the topology depicted in Figure 8.

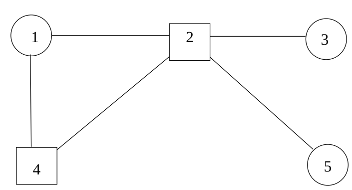

Figure 8: A simple topology to illustrate the branch and bound algorithm

To obtain the worst case, we have to compute with nodes having a maximal number of adjacent nodes. The maximal degree of the nodes in a graph G is equal to $n-1$ (case of a complet graph). Since it is not possible to return to the source, the number of valid adjacent nodes of an arbitrary intermediate node in a hierarchy is limited by $n-2$. The maximal number of the adjacent edges starting from the leaves of a hierarchy having k leaves is trivially $k \cdot (n-2)$. So, the number of the successor hierarchies in the search tree of a hierarchy having k leaves is at most $2^{k \cdot (n-2)}$.

The number of leaves in a hierarchy at the level i of the search tree is limited by:

$$fm_i = (n-1)(n-2)^{i-1}$$

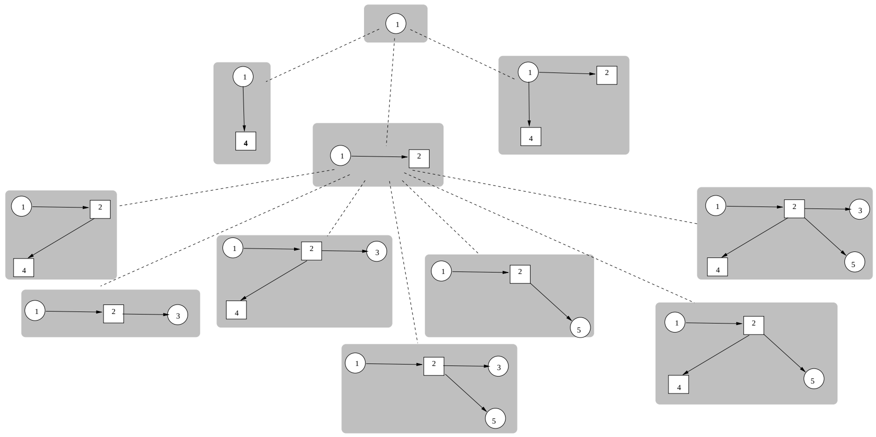

Figure 9: The search tree after the first three steps

A bound on the number of hierarchies at the level i can be given by the recursive relation:

$$B_i = B_{i-1}(2^{(n-1)(n-2)^{i-2}\cdot(n-2)} - 1) = B_{i-1}(2^{(n-1)(n-2)^{i-1}} - 1)$$

such that

$$B_0 = 1$$

.

The solution of this recursive relation is

$$B_i = B_0 \prod_{j=1}^{i-1}(2^{(n-1)(n-2)^j} - 1)$$

Taking into account Lemmas 3 and 4, the maximal length of an itinerary from the source to an arbitrary destination leaf is $l_{max} = p + (n-p)\cdot d$.

The maximal number of hierarchies in the search tree with depth limited by l_{max} can be computed by summing the hierarchies at each level of the search tree:

$$n_{max} = \sum_{i=0}^{p+(n-p)\cdot d-1} \prod_{j=1}^{i-1}(2^{(n-1)(n-2)^j} - 1)$$

This value corresponds to a large upper bound of the maximal number of the developed nodes in the search tree using the branch and bound algorithm. Let us notice that the algorithm stops very rapidly if the node degree is maximal as it is supposed in the above worst case execution time calculus. In the complete graph corresponding to the maximal degrees, the optimal spanning hierarchy is a star, in which each destination is directly connected to the source and this hierarchy (tree) is trivially the shortest one. The algorithm already finds this result in the first step between the successor hierarchies of the source.

3.1.2 Optimal Solution in a large but limited hierarchy

To find the optimal solution of the Steiner problem, very simple *tree enumeration algorithms* was proposed by Hakimi and Lawler in [9] and in [10]. The basic idea of these algorithms is the computation of a sub graph generated by a sub set of the possible Steiner nodes and then the computation of the minimal length spanning tree in the generated sub graph (this latter algorithm is feasible with polynomial time but the enumeration of sub sets of Steiner nodes implicates exponential time execution). Unfortunately, the same idea of construction of a minimal spanning hierarchy in a graph with the here given constraint is not trivial. The enumeration of potential nodes is not trivial because nodes can be used several times.

In the place of the enumeration of all possible (partial) hierarchies, we propose an other exact algorithm based on the limitation of the search graph. This proposition called Largest Possible Hierarchy Construction (LPHC) algorithm is based on the following three phases:

- construction of the largest directed hierarchy H_{max} from the source node in the graph G according to the given constraints
- transformation of the directed hierarchy to a directed graph by adding a virtual destination for each destination connected to the occurrences of the same destination in H_{max} using zero cost arcs
- search a directed minimal Steiner tree spanning the source node and the virtual destinations in this latter directed graph.

The *largest hierarchy* can be built from the source using a classic search algorithm: for example the depth first search algorithm. At each iteration of the search algorithm the adjacent nodes of the current node are enumerated. If an adjacent arc of the current node is already used in the itinerary (cf. Constraint 2) or the adjacent node corresponds to the source or the node is a splitter node and used in the itinerary (Lemma 3) or the node is not a splitter but it is used d times (Lemma 4) then the adjacent node is not added to the hierarchy else the successor node is connected to the hierarchy using an arc directed to it. The result of this depth first search a hierarchy H_{max} with limited radius. Each itinerary of H_{max} satisfies the constraints and has a maximal length (*i.e.*, the successors of the leaves do not correspond to the given constraints).

Figure 10 contains a simple network topology. Let us suppose that the DMPSH from the source node 1 to the destinations 6 and 7 is needed. The result H_{max} of the depth first search is illustrated in Figure 11. Corresponding to the constraints, a splitter node belongs at most once and a node which can not split belongs at most twice to a same itinerary. An itinerary can not contain a same edge twice.

The *transformation of the directed hierarchy to a directed graph* processes two operations. Since the DMPSH under the given constraints is a minimal length hierarchy, the branches

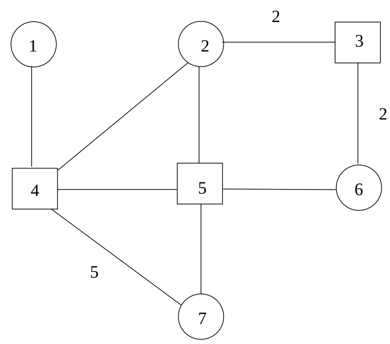

Figure 10: The network topology to illustrate the largest hierarchy

of H_{max} which not lead to a destination are not present in the optimal solution. At a first step, the transformation eliminates all leaves which do not correspond to destination nodes. This step is illustrated in Figure 12.

The searched DMPSH corresponds to the minimal length sub hierarchy in H_{max} routed at the source node and spanning the destinations. In this optimal sub hierarchy, each destination must be spanned once but it can belong to H_{max} several times. One of its occurrences should belong to the optimal hierarchy. To insure the equivalence of the different occurrences of a same destination node, one fictive node corresponding to each destination is added to H_{max}. All occurrences of a same destination are connected to the corresponding fictive node using an arc directed to the fictive node with cost zero. In this way a directed graph G_d from the source to the fictive destinations is obtained as it is illustrated in Figure 13.

The last phase is *the search of the directed minimal Steiner tree* spanning the source and the fictive destinations and corresponding to the given constraints in G_d. By construction, the constraints imposed by Lemmas 3 and 4 and expressed by Constraint 2 are respected in all sub hierarchies of H_{max} and in this way also in G_d. Contrarily, Constraint 8 which takes into account the splitting capability of the nodes is not respected by the previous phases. So, the Steiner tree search algorithm should be a modified Steiner tree search. Traditional exact tree computation algorithms as Hakimi's [9], Lawler's [10] or Balakrishnan-Patel's[11] algorithms can be used adding a supplementary selection criterion to find the optimal directed tree: only trees corresponding Constraint 8 can be accepted.

Theorem 2 *The LPHC algorithm finds the optimal solution.*

Proof. Let us suppose that the algorithm LPHC provides a solution H^*. Let us suppose that this solution is not optimal. In this case, there is a hierarchy H' spanning the destinations with less length. Since H' is the optimal solution, it corresponds to the criterion given by Constraint 2, 8 and Lemmas 3, 4. So $H' \subset H_{max}$ by construction. In H_{max} the

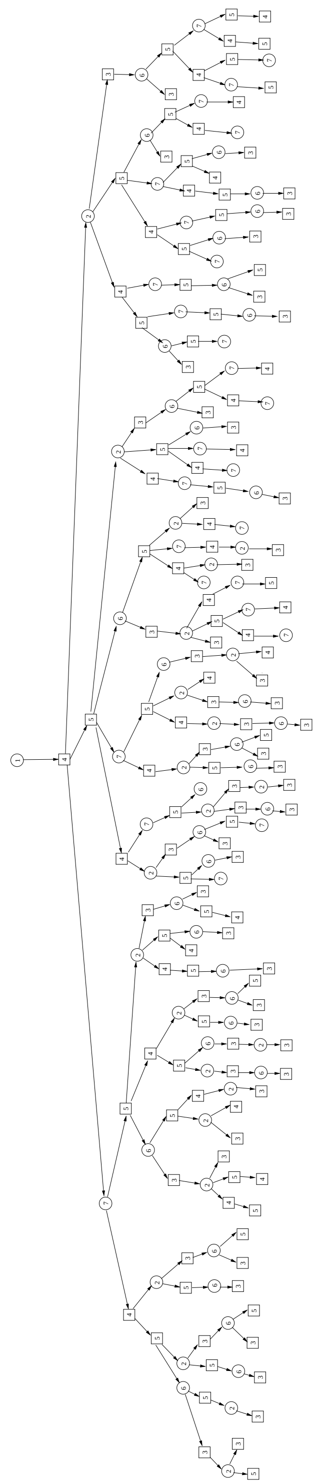

Figure 11: The largest hierarchy H_{max}

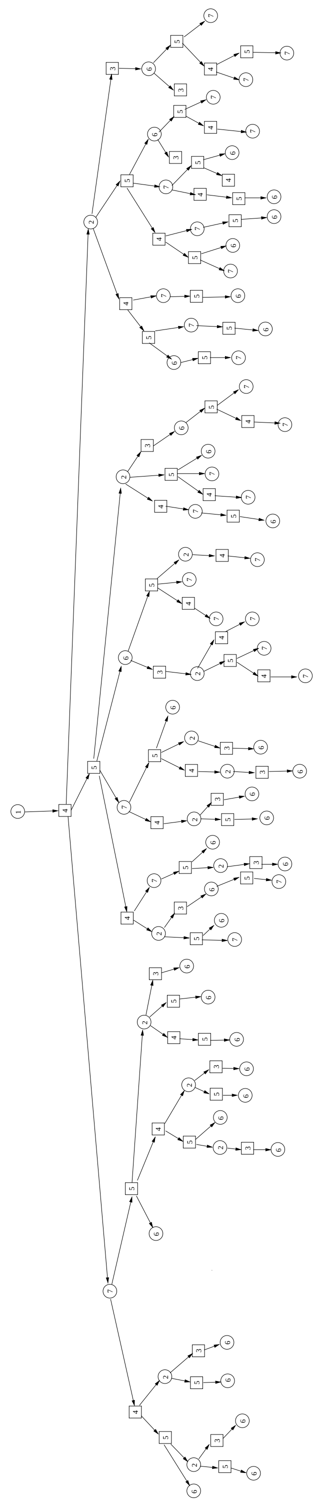

Figure 12: The pruned hierarchy

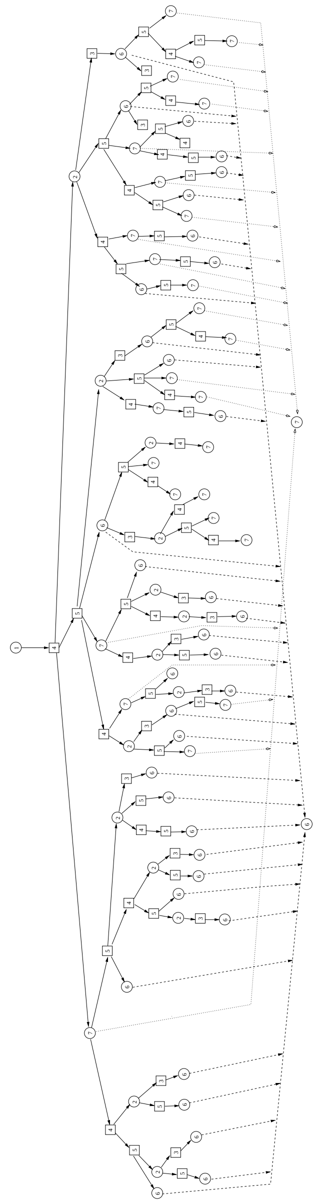

Figure 13: The completed directed graph from the source to the fictive destinations

minimal cost directed Steiner tree returned by the algorithm is H^*. The contradiction is trivial.

Let us suppose that the LPHC does not admit solution but the solution H^* exists. In the same way one can demonstrate that $H^* \subset H_{max}$. H^* corresponds to all criterion, also to Constraint 8. So the exact Steiner tree construction algorithm with Constraint 8 should find this sub graph in the last step of the algorithm. ■

On the complexity of the LPHC algorithm

A large upper bound of the proposed algorithm can be done as follows.

In the worst case, the degree of the nodes in G is limited by $n-1$. Applying the result of Lemmas 3 and 4, the maximal hop length of an itinerary from the source to an arbitrary destination is limited by $p + (n-p) \cdot d$. A large upper bound on the number of nodes in the largest possible hierarchy is given by $\tilde{n} = (n-1)^{p+(n-p)\cdot d+1} - 1$.

Since the number of leaves is limited by $(n-1)^{p+(n-p)\cdot d}$. the complexity of the first two steps of the construction resulting the directed graph is in
$O((n-1)^{p+(n-p)\cdot d+1})$.

The maximal number of branching nodes $\tilde{b}$ in the worst case is equal to the the number of not leaf nodes: $\tilde{b} = (n-1)^{p+(n-p)\cdot d} - 1$. The Steiner tree computation using the Lawler or the Hakimi Steiner tree enumeration algorithm implicates a complexity $O(2^{\tilde{b}}\tilde{b}^2)$.

Note There are several possibilities to compute the exact solution. For example, a different mathematical formulation of the optimal multicast routing can be found in [12]. In this paper, the multicast routing problem with and without wavelength converters in WDM networks is analyzed. The authors search the possibilities of multicast routing both in opaque networks (using optical/electronic/optical O/E/O conversions) and transparent networks (using all optical solutions) with limited splitter fanout (splitting degree) of nodes. They expand the work also to fractionated multicast traffic (with traffic grooming/aggregation in the ingress points). The basic formulation of the problem turns out to be a mixed integer linear program and it is considered - as a special case of the directed Steiner problem - NP-complete (cf. [13]). The fundamental difference between the given MILP formulation in [12] and our recent study is the following. The MILP formulation suppose multicast trees. But - according to our analysis - *the tree constraints are not necessary to construct optimal directed spanning structures.* We do not investigate here the MILP formulation of the optimal structure. This formulation and the corresponding research of the optimal solution using traditional solvers as CPLEX can be interesting, future works in the domain.

3.2 Heuristic solutions

3.2.1 Some earlier proposed algorithms

The hardness of the multicast routing in WDM networks with sparse light splitting capability has been discussed in several papers and different algorithms have been proposed. Source based and constrained WDM multicast routing has been analyzed in [8] and four well known heuristic algorithms have been proposed to resolve the routing. The authors suppose that the splitting and wavelength conversion capability of the nodes is variable and multicast capable nodes are spares. The discussed routing problem corresponds to the case where only the constraint on the node degree should be taken into account. "Re-route-to-Source" and "Re-route-to-Any" algorithms begin with the construction of an arbitrary multicast tree (a partial spanning tree which may be a shortest path tree or an approached Steiner tree for example). Since branching nodes of the multicast tree do not always correspond to multicast capable nodes of the physical network topology, then the tree is hashed to a set of trees (called "light forest"). The hash of theses trees is realized taking into account the splitting capability of the nodes. In the case of a conflict in the algorithm "Re-route-to-Source", the first node toward the source having splitting capability is used as virtual source. The algorithm "Re-route-to-Any" permits to connect a sub-tree to a splitting capable node anywhere in the multicast forest. "Member-first" and "Member-only" heuristics compute a multicast forest tree by tree, one tree at a time, adding the members to the structure one after the other. "Member-first" algorithm enhances a tree using a priority queue of adjacent edges and with preferring edges toward members. "Member-only" heuristic construct trees with a Takahashi-Matsuyama like algorithm: the closed member is added to the tree using a shortest path to a multicast capable node. If it is not possible, a new tree is started from the source. Let us notice that these heuristics always produce a set of trees rooted at the source node and do not allow cycles to construct more advantageous hierarchical structures.

In [14] the authors deal also with the constrained multicast routing problem with sparse multicast (splitting) capability of switches. They define the routing and wavelength assignment (RWA) problem in a directed graph representing the optical network. An RWA is a directed tree-like spanning structure considering the wavelength availability on the used arcs and the multicast capability in the nodes. Similarly to the results of the previously mentioned heuristics, the RWA structure is a hierarchy without cycles in the original topology graph. The authors proof that finding an RWA in the general case is NP-complete.

In the paper [15] optimal virtual topologies for multicast communication in WDM networks is discussed. A virtual topology corresponds to a set of lightpaths using different wavelengths to avoid conflict if a same physical link is shared by two or more lightpaths. The extremities of the lightpaths (which are switches) can convert wavelengths arbitrarily. The optimal one-to-many virtual topology design problem consists to find the set of lightpaths spanning the given multicast group with minimal value of maximal (or average) hop distance from the source to the destinations. The authors state that this problem is NP-complete. Furthermore they prove that there is an optimal virtual topology containing only one directed virtual path from the source to each destination if any virtual topology exists for the group. With our terminology, the optimal virtual topology can be seen as a hierarchy disassembled into a set of lightpaths. The same idea is developed for arbitrary multicast tree topologies in [16]. A first problem proposed by the authors corresponds to finding a multicast virtual topology (a set of lightpaths) covering a given multicast tree under some constraints. The switches of the network are supposed "tap and continue" type switches. That is each switch can tap the message for local utilization as it passes. Since the tap operation consumes energy, the number of switches taping the same signal is limited along a lightpath. The endpoint (the last destination which can be served) of a lightpath can convert the signal to the electrical domain and reinsert it to begin a new lightpath if needed. The objective is to compute a set of lightpaths covering the destinations minimizing the maximal virtual hop distance (the number of successive lightpaths) using a limited number of tap operation in each lightpath and a limited number of wavelengths per fiber. This first problem is traitable in polynomial time. Figure 14/a) presents a multicast tree. Let us suppose that there is two wavelength available in the fibers and only one intermediate tap operation authorized in intermediate nodes of a lightpath. Supposing that all nodes are destinations, the optimal virtual topology is presented in Figure 14/b). Figure 14/c) illustrates the fact that this topology corresponds to an optimal hierarchy.

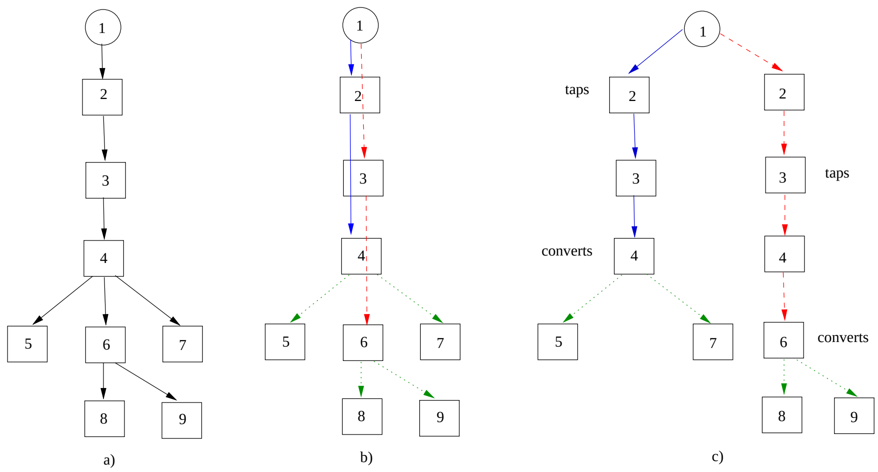

Figure 14: Example of optimal virtual toplogy with at most two wavelengths per fibers and one tap operation.

In their second problem the authors propose the use of light-trees instead of lightpaths to span the given multicast tree. They suppose that the number of destinations which may tap the message in a light-tree is limited. This kind of decomposition is called splitting virtual topology. Contrarely to optimal tap-and-continue topology computation, the problem of the optimal splitting virtual topology is NP-complete [16]. Let us notice that in both cases the thought is the decomposition of the destination set applying lightpaths and light-trees using strictly the arcs of a given multicast tree. So, the resulted virtual topologies are not necessary optimal. For a given network topology and a (s, D) set, the optimal solution (the hierarchy with minimal diameter taking into account the constraints) does not always correspond to a tree and naturally can be different from an arbitrary multicast tree. Finally, we point to the paper [17] of the same laboratory in which the virtual topology design of multicasting is generalized for the cases where multiple potential sources (originators, mirrors of video sources, etc.) are available in the WDM network. To find a virtual topology from any source to all of the destinations, the number of available wavelength par fiber and the number of transmitters in the switches are limited and there is no wavelength conversion possibility in the network. Algorithms for path and ring topologies are proposed. The effect of the objective function on virtual topology design is discussed in [18] From our point of view, the generalized virtual topology design problem can be analyzed successfully using hierarchies and this can be with different objective functions. For simplicity, our main objective in this section to find "good" multicast routing structures (with less length) in WDM networks where the nodes are not always multicast capable nodes.

The multicast tree construction in sparse splitting capable WDM networks was discussed also in [19]. The heuristic proposition concerns a two phased algorithm: at a first time a shortest path tree is computed and then the tree is hashed into a set of trees taking into account the splitting capability of the nodes. Two modified Steiner-tree heuristics (modified Kuskal's and Takahashi-Matsuyama's algorithms) were proposed in [20] to construct partial spanning trees. The performance of the second proposed solution is better than the performance of the Member-Only heuristic. These propositions always concern tree constructions and do not aim with generalized hierarchies.

Beside the MILP formulation of the optimal (minimal cost), tree-based multicast under splitting constraints, the paper [12] proposes the use of Takahashi-Matsuyama's heuristic to compute 2-approximated solutions for multicasting in WDM networks with wavelength converters. In this cases, the optimal (minimal cost) solution corresponds trivially to a Steiner tree. If there are no wavelength converters in the network (which case corresponds to our study), the authors propose an other heuristic. This heuristic works in a layered graph, which is a graph with a layer for each wavelength and the edges of a layer reflect the availability of the given wavelength in the different fibers. The proposed algorithm compute (if it is possible) an approximated Steiner-tree in each layer using the Takahashi-Matsuyama's heuristic. Then the spanning tree with the minimal cost is selected from the different layers. As these heuristics built 2-approximated solutions for the related Steiner problem, these obtained solutions are not necessary 2-approximations of our optimal spanning hierarchy problem.

3.2.2 A hierarchy construction heuristics

To find an approximated solution of the well known Steiner problem without constraints, one of the more simple and efficient heuristic is the algorithm proposed by Takahashi and Matsuyama [21]. This heuristic adapt the idea of the exact Prim's algorithm for MST construction to find 2-approximated PMST. The well known Member Only heuristic (cf. in [8]) for WDM multicast routing adapt this heuristic to the case of sparse splitting capable switches, as it is known. This heuristic suppose helplessly that the needed routing structure is a tree (or a forest). The strong tree construction is controlled by a useless constraint in the proposed algorithm:

- a new member (a new destination) is connected to the tree under construction using a path, if the node of attachment is a multicast capable node, a leaf or the source and there is no intersection (an other common node) between the tree and the path. In other world: if the result corresponds to a tree.

A slight modification of the Member Only heuristic allows to construct more efficient hierarchical routing structures. The only one usefull constraint to connect a new member to a hierarchy can be formulated as follows:

- a new member is connected to the tree under construction using a path, if the node of attachment is a multicast capable node, a leaf or the source and any arc of the directed hierarchy is not used by the (directed) path.

So, the resulted set of directed hierarchies can be shortest than the forest constructed by the original Member Only algorithm. Figure 15 illustrates the difference between the two algorithms. To simplify, let us suppose that the network topology corresponds to the depicted graph, the source is the node 1 and there are two destinations: nodes 9 and 10. Using the Member Only heuristic, since node 4 is not capable to split, two tree are built for the two destinations and consequently two wavelengths should be used in the common edges. The total length of this light forest is equal to 11 (expressed with the number of hops or optical channels). In the given topology, the proposed modification results a hierarchy corresponding to one of the two possibles depicted also in the figure. The modified heuristic constructs shortest hierarchies; the length of both solutions is equals to 10. Let us notice that the hierarchies are better than forests forasmuch they use only one wavelength.

Algorithm 2 The draft of the proposed modified Member Only heuristic

Input: the graph $G = (V, E)$, the set MC of splitting capable nodes, the source s, the destination set D
Output: F_H, a set of directed hierarchies rooted at s
  while $D \neq \emptyset$ **do**
    $H \leftarrow s$ {initialize a hierarchy with the source}
    $V \leftarrow \{(s\}$ {set of the MC nodes and leaves in H}
    select $d \in D$ the closest destination which can be connected to H
        {the connection should correspond to the new criterion}
    while d exists **do**
      $P \leftarrow$ *path_between(d,H)*
      $MC_P \leftarrow$ *MC_nodes_in(P)*
      $V \leftarrow V \cup MC_P \cup \{d\}$
      $D \leftarrow D \setminus \{d\}$
      select $d \in D$ the closest destination which can be connected to H
    end while
  end while

3.2.3 Perspectives

For simplicity reason, only the case of multicast routing in WDM networks without wavelength converters and with sparse splitting capable switches is analyzed in this paper. Moreover, we supposed that the O/E/O conversions are undesirables. The objective is to minimize the overall cost; so we are interested to find DMPSH under the mentioned conditions.

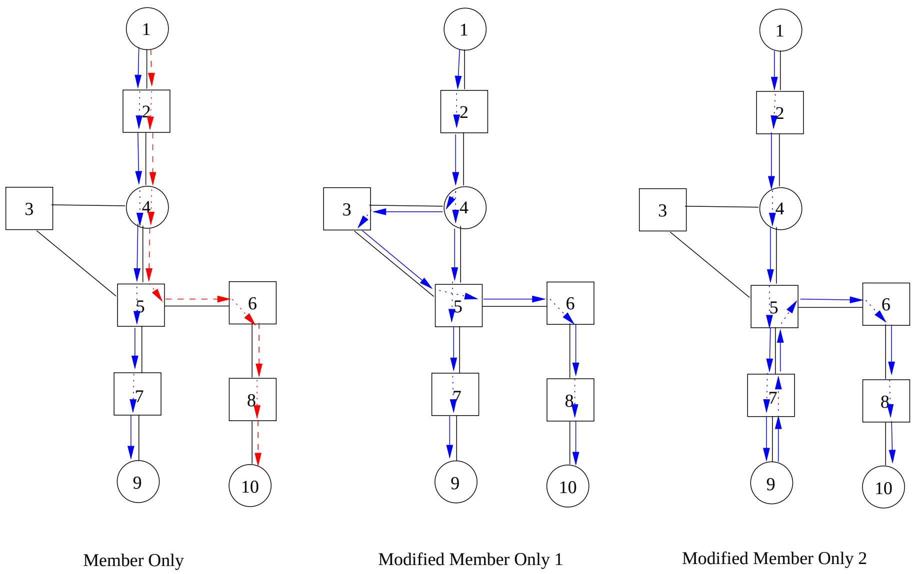

Figure 15: The forest constructed by Member Only and the DMPSH constructed by the proposed heuristic.

In future optical networks, multicast routing should be performed in various cases with different conditions. The optimal hierarchical solution of the different multicast routing cases depends strongly on the constraints determined mainly by the physical capacity of switches and the spatial distribution of the different switches. Future works should analyze the case of WDM networks with sparse wavelength converters and splitters, with sparse virtual sources which contain both converters and splitter units. An interesting challenge consist to add constraints on the diameter of routing structures since the propagation of the light without amplification is limited. In optical networks without wavelength converters and splitters, the multicast should be performed using lightpaths and permitting the O/E/O conversions in some nodes. Since the O/E/O conversion and the handling in electrical domain add important latencies, The aim is the minimization of the conversions (the number of the branching nodes) in the multicast routes Both the optimal solutions and the heuristics vary from one case to the other. In our future activities, the analysis of different usefull cases is imagined.

Network access providers encounter with two kinds of routing problems. In the first, the routing requests are handled one after the other and the network manager should find a route (tree or other) immediately when a request is presented: one talk about *on-line* routing problem. In the second case, the expected traffic matrix is known and the network manager should establish a system of routes in a *off-line* manner. It will be interesting to examine how the hierarchy based multicast routing algorithms can improve the overall performance of the network management.

4 Partial minimum spanning hierarchies with end to end constraints

The second example to illustrate the interest of hierarchy like structures in graphs corresponds to the multi-constrained optimal multicast routing. In IP based networks several multicast routing techniques have been proposed for point-to-multipoint communications leading to a reduction in network resource consumption. Recent, typical multicast applications require the satisfaction of QoS criterion. Several multimedia applications which are often based on multicasting needs to satisfy more than one criterion that's why multi-constrained QoS routing is asked for them. Various real-time services, like TV, audio- and videoconferencing, shared document edition and telemedicine are being deployed over the Internet. Often, the requirements of these applications is specified in terms of the QoS metrics like desired bandwidth, end to end delay, variation of the delay, response time, tolerated packet losses etc. Routing algorithm s in the network layer have a critical role to play in the QoS provision process. It should provide the required QoS by considering the QoS metrics in the route selection process.

Compared to the well known Minimum Steiner Tree problem which is based only one cost metric and which is known to be NP-complete [4], the construction of trees or sub graphs satisfying multiple QoS requirements is more complex. Multi-constrained QoS Routing even in the unicast case is known to be a NP-complete problem. In the *Multi-Constraint Path (MCP) problem* (cf. [22]), a path P from the source node s to the destination d is wanted such that the QoS constraints are respected: the sum of each additive metric should be less then a given threshold. This kind of paths is called *feasible* path. In the *Multi-Constraint Optimal Path (MCOP) problem*, in addition to find a feasible path, a path $P*$ with minimal length is wanted. The definition of the "length" of a path is not trivial. The mostly used length function are presented in the next sub section.

In multi-constrained multicast routing cases sub graphs with feasible paths between group member pairs are wanted. A good start point of the problem and a clear classification of the used metrics and algorithms can be found in [23]. In the following, we resume the most important models of *single source* multi-constrained multicast routing problems.

4.1 Multi-constrained multicast routing problems

In contrast to multicast routing based only on one metric, where the routing structure corresponds to a tree, in multicast QoS routing with several constraints the feasible routing structures are not always trees. A feasible routing structure corresponds always to a connected directed sub graph but the cyclomatic number of this sub graph can be different from 0. So, it is not always a tree and as we shall see next, often, it does not correspond to a set of directed trees. In any case, it corresponds to a directed sub graph $G_M = (W, H)$ containing at least a feasible path from the source to each destination. It is always true that this sub graph corresponds to a hierarchy (cf. Lemma 5 later).

Since the direction in which the edges of the topology graph are used in the routing structure is important, we propose to discuss the problem using directed concepts as directed sub graphs, paths, trees, hierarchies.

With the claim of synthesis, three multi-constrained multicast routing problems have been formulated in [24]. The authors suppose that the topology graph $G = (V, E)$ is given, s is the source node and D corresponds to the set of destinations in V. An m-dimensional vector $\overrightarrow{w}(e) = [w(e)_1, ..., w(e)_m]^T$ of additive type metric values is associated to each edge $e \in E$. The end to end constraints (constraints from the source to the destinations) are also given by a m-dimensional vector $\overrightarrow{L} = [L_1, ..., L_m]^T$.

Since the metrics are additive, an end to end weight vector can be associated to a path P in G as follows:

$$\overrightarrow{w}(P) = \sum_{e in P} \overrightarrow{w}(e)$$

The multi-constrained multicast routing problems have been formulated as follows.

Multiple Constrained Multicast (MCM)

In this first case, the goal is to find a directed sub graph $G_M = (W, H)$ in which there exists a directed path $P(s, d_j)$ from the source s to each destination $d_j \in D$ such that:

$$w_i(P(s, d_j)) \leq L_i \text{ for } i = 1, \ldots, m$$

or by using the Pareto dominance:

$$\overrightarrow{w}(P(s, d_j)) \overset{d}{\leq} \overrightarrow{L} \tag{5}$$

Since, the sub graph or the routing structure solution of the MCM problem must fulfill (5), a trivial solution can be found by composing a set of feasible paths: one feasible path from the source to each destination if such a path exists. This set of paths may contain cycles. These cycles can lead to helpless redundancies since some cycles can be reduced without less the criterion (5).

Multiple Parameter Steiner Tree (MPST)

In this case, a directed sub graph $G_M = (W, H), \{s, D\} \subseteq W$ is wanted such that a length function $l(G_M)$ is minimal.

In [24] the length of a sub graph G_M corresponds to a scalar metric proposed for multi-constrained unicast routing and defined as:

$$l(G_M) = \max_{i=1,\ldots,m}(\frac{w_i(G_M)}{L_i}) \tag{6}$$

where $w_i(G_M) = \sum_{(u,v)\in H} w_i(u,v)$ the metrics being additive.

This non-linear length function characterizes very well the satisfaction of the constraints on a path P. If all the constraints given by the vector $\overrightarrow{L}$ are satisfied on the path P, then this value $l(P) \leq 1$. So, a feasible path can be defined as a path with length less then 1.

Note that for multi-constrained multicast routing the tree T with minimal length $l(T)$ is irrelevant since the constraints are given for paths from the source to the destinations and not for the overall set of edges. An overall length of the spanning tree does not characterize the satisfaction of the multiple objectives in the destination nodes.

Multiple Constrained Minimum Weight Multicast (MCMWM)

This case is the combination of the two previous one. The goal is to find a sub graph $G_M = (W, H)$ in which there exists a directed feasible path to each destination $d_j \in D$ and the length of the sub graph is minimal:

$$\overrightarrow{w}(P) \overset{d}{\leq} \overrightarrow{L} \text{ and } l(G_M) \text{ is minimum.} \tag{7}$$

For the previously mentioned multi-constraint multicast routing problems several exact solutions (which work in particular cases) and heuristic algorithms (for more general route computations) have been proposed. The reader can find some related works in the next section. From the point of view of the generality of the solution, the most interesting algorithm is the Multicast Adaptive Multiple Constraints Routing Algorithm (Mamcra) which works in the case of additive metrics [24]. This algorithm is based on the fact that if there is a feasible solution for the multi-constrained multicast routing problem, then the set of the "shortest paths" using the length function 6 from the source to the destinations is a feasible solution. So in the first step of Mamcra, a multi-constrained shortest path is computed from the source node to each destination using a unicast QoS routing algorithm, Samcra [25]. This set of paths is not optimal from the point of view of resource allocation and redundancies. It can contain sub graphs which are also feasible under the given constraints and which are more preferable considering the cost (the number of edges or the total length for example). Since the set of obtained paths can contain cycles, it is then optimized by the second step of Mamcra to obtain a multicast sub graph that uses as few links as possible. This second step is performed using a greedy algorithm to eliminate a part of helpless redundancies but without guarantee on the elimination of all helpless redundancies.

Let us notice that the "optimal" solution of the multicast routing does not belong obligatory to the graph involved by the set of shortest paths (using an arbitrary cost or length function). The helpless redundancies in the solution computed by Mamcra can be important.

How the minimal length routing structure can be defined? To obtain more preferable routing structures, a more appropriate definition of the multi-constrained multicast routing problem is needed.

4.2 Problem formulation

In the case of IP based multicast routing, a multicast group is identified by an unique multicast address and generally the routing structure is a tree. Using a tree only one address is needed for a group (there is no conflict in the nodes).

Multi-constrained multicast routing can not be resolved using only one tree (and so using only one multicast IP address). To prove this, a very simple example of Figure 16 with one source and two destinations can be considered. The two-dimensional link values are also indicated in the figure. Let us suppose that the end to end QoS constraint vector for the destination nodes 5 and 6 is $\overrightarrow{L} = (5, 5)^T$. In this case, there is only one feasible path to each destination. Trivially, the only one feasible (and so optimal) sub graph contains a cycle and a unique address can not be used because of the data forwarding conflict in the node 4.

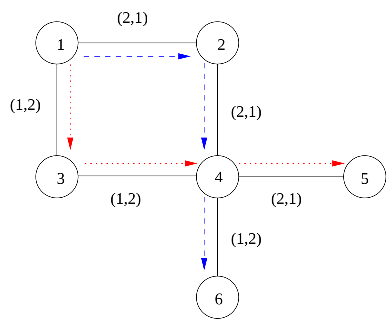

Figure 16: Two trees (paths) are needed to satisfy the constraints

Let us suppose that the multicast communication can be realized using multiple addresses or different labels applying the MPLS technology (cf. [26]) or applying an explicit multicast routing (cf. [27]).

The aim of the routing which we propose here is to find a directed sub graph containing at most a feasible directed path from the source to each destination and minimizing a cost function. In fact, real multicast routing politics aims the minimization of allocated network resources. The simple length of the structure (the hop number) seems the best cost functions. To simplify, we propose the hop number $h(G_M)$ of the sub graph G_M as cost to minimize but different linear (additive) cost functions may correspond to the optimization goal. Using this objective function the problem of Multiple Constrained Minimum Length Multicast (MCMLM) has been presented in [28]. Since our paper aims the analysis of this latter optimization, we present the problem as follows.

Problem 2 *Minimum directed spanning sub graph with end to end constraint on the paths.*

Let $G = (V, E)$ be a undirected graph valuated with a vector $\overrightarrow{w}$ of m positive values on the edges, $s \in V$ a source node and $D \subset V$ a set of destinations. The link values are from additive type metrics. Let $\overrightarrow{L}$ be the vector of the end to end constraints.

The problem corresponds to finding the directed graph G_M^* with minimal hop length $h(G_M)$, spanning $\{s\} \cup D$, having at least a directed path $P(s, d_j)$ from the source to each destination $d_j \in D$ with respect the given constraint vector $\overrightarrow{L}$:

$$\overrightarrow{w}(P(s, d_j)) \overset{d}{\leq} \overrightarrow{L} \text{ and } h(G_M) \text{ is minimum.} \tag{8}$$

Lemma 5 *The sub graph G_M^* corresponding to the optimal solution of Problem 2 is a directed hierarchy (which does not correspond always to a tree).*

Proof. Similarly to the proof of Lemma 1, it is easy to see that the optimal spanning structure is a directed hierarchy (there is no two incoming itineraries for a given node occurrence in the hierarchy). The optimal hierarchy does not correspond obligatory to a partial spanning tree. Trivially, in this solution an edge of the original graph can be used several times and both in the two directions: a simple example is shown in Figure 17. In this example the source is the node 1 and the destinations are the nodes 6, 7 and 8. A two dimensional weight vector is associated to each edge as it is depicted in the figure. With the given values there is only one feasible directed path from the source to each destination. These feasible paths are also indicated in the figure. The unique feasible hierarchy and in this way the minimum directed spanning hierarchy uses the edge $(4, 5)$ three times and in the both directions. ■

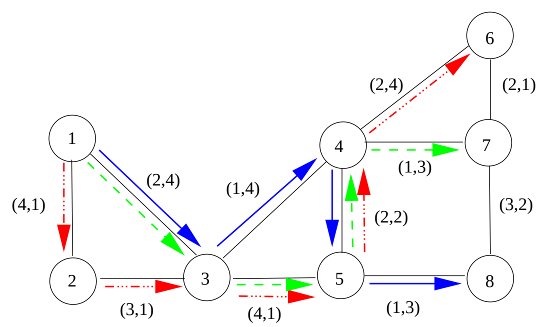

Figure 17: Several itineraries can use the same edge

We propose to call the optimal solution G_M^* of Problem 2 as directed minimum partial spanning hierarchy (DMPSH) under end to end constraints.

Lemma 6 *Let G_M^* the DMPSH in the graph G under the end to end constraints given by $\overrightarrow{L}$. Let r be an arbitrary node in G_M^* and $H_r \subseteq G_M^*$ be the hierarchy rooted in the node r.*

1. *The hierarchy $H_r \subset G^*_M$ rooted in the node r is a DMPSH.*

2. *Let us suppose that H_r is a DMPSH under the constraint $\overrightarrow{L_r}$ from r to the leaves. A sub hierarchy H^r_i (if it exists) having the node r_i as root corresponding to a child of r is also a DMPSH under the the following constraints:*

$$\overrightarrow{L_{r_i}} = \overrightarrow{L_r} - \overrightarrow{w}(r, r_i)$$

Proof. Trivially, H_r is a hierarchy. If it would not be true, then G^*_M could not be a hierarchy and so a DMPSH. H_r is a DMPSH under end to end constraints from the root r to its leaves. Let us suppose that this is no the case and H_r is not a minimal solution but only a feasible solution for the partial problem. In this case, there is a DMPSH $\tilde{H}_r$ rooted at r the length of which is less then the length of H_r Replacing H_r by $\tilde{H}_r$ in the hierarchy G^*_M we obtain a feasible spanning hierarchy with less length which is in contradiction with the fact that G^*_M is a DMPSH. The contradiction is obvious.

Since the metrics are positive, additive metrics, the most light constraints which should be satisfied by all of the sub hierarchies of r is given by $\overrightarrow{L_{r_i}} = \overrightarrow{L_r} - \overrightarrow{w}(r, r_i)$. Let us suppose that this vector constraint is not respected by the sub hierarchy rooted at r_i. In this case, there is at most an itinerary from r to a leaf which can not satisfy the constraint $\overrightarrow{L_r}$ from r to the same leaf which is incompatible with our hypothesis. H^r_i should be a DMPSH with the end to end constraint $\overrightarrow{L_{r_i}}$. Because H^r_i is a sub hierarchy of G^*_M rooted in r_i, it is a DMPSH as it was proved. ■

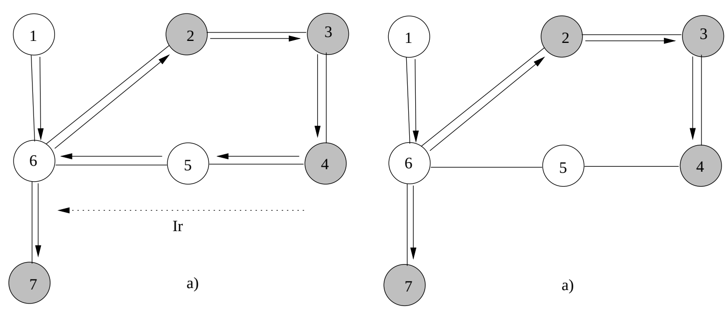

Figure 18: A node belongs at most once to an itinerary of a DMPSH

Lemma 7 *In a DMPSH G^*_M under end to end constraints in a given graph $G = (V, E)$, a node $n \in V$ belong at most once to each itineraries of G^*_M.*

Proof. Let us suppose that the node n is used twice by an itinerary I in the DMPSH. Let D_b the set of destinations in the itinerary between the two occurrences of n and D_a the destinations after the second (last) occurrence of n. Let r the last node in D_b taking into account the direction of arcs in G^*_M and I_r be the itinerary from r to n. The case is illustrated in Figure 18. The itinerary I_r can be removed from I without less the connectivity of the destinations in D_b and in D_a to the source. The resulted structure is a (partial) spanning hierarchy. Since the metrics are positive, additive metrics, the reduced itinerary is shortest in all metrics then I.

So an itinerary of a DMPSH under end to end constraints can not contain the same node twice ■

Another illustration can be found in Figure 19. There are no destinations between the two occurrences of the node n. Trivially, the cycle (n, j, a, b, n) can be removed from this itinerary without less the connectivity of the eventual destinations being after n.

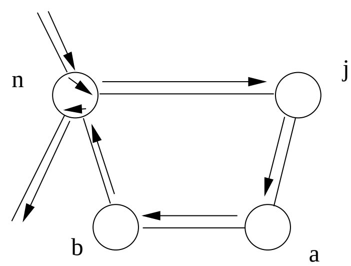

Figure 19: Deletion of a helpless cycle from an itinerary

Consequently, the edges of G are used also only once on a same itinerary of a DMPSH.

4.3 Research of the optimal solution

MCM, MSTP and MCMWM problems formulated in [24] are NP-hard problems. In Problem 2 the objective function of the spanning structure is lightly changed compared to MSTP: instead of the non linear length function an additive length (the hop number) should be minimized by the solution. This modification does not change the difficulty of the problem.

Lemma 8 *Problem 2 is NP-complete even if the group contains only one destination..*

Proof. When there is only one destination to span, a feasible path from the source to this destination is wanted. In the worst case, there is only one solution corresponding to the "shortest" path using the non linear length function mentioned here before. The search problem of this shortest path is known as a NP-complete problem [29]. ■

In the following, we propose a simple exhaustive search algorithm based on the branch and bound algorithm.

4.3.1 Branch and bound algorithm

Applying the branch and bound algorithm, directed partial spanning hierarchies corresponding to the limitation given by Lemma 7 can be enumerated. The algorithm selects hierarchies in increasing order of total length (number of hops). The first hierarchy of the enumeration covering all destinations and satisfying the end to end criteria corresponds to the optimal solution. For practical reason, each node of the search tree contains a hierarchy and its length. In the leaves of the hierarchies, the end to end values on the itinerary from the source to the leaf are also known. A hierarchy is not feasible, if it contains a leaf which does not satisfy the given constraints. If there is no leaf in the search tree satisfying the end to end criteria, then the problem has no solution in the given graph with the imposed constraints.

The successors of a hierarchy H in the search tree are the hierarchies enlarged only by adjacent arcs from the leaves of H (the used links are called fringe links in some papers). To enumerate all the successors of a given hierarchy in the search tree, the not empty sub sets of the adjacent arcs should be enumerated exhaustively. This enumeration ensure that a hierarchy is visited only once by the algorithm. The computation of the set of successors of a given hierarchy H_{min} is detailed in the following algorithm:

Algorithm 3 Computation of successor hierarchies in the search tree

Input: the valuated graph $G = (V, E)$, the hierarchy H_{min}
Output: S the set of successor hierarchies regarding end to end QoS requirement and Lemma 7

```
S ← ∅
A ← ∅          {set of possible adjacent arcs}
for all leaf n of H_min do
  A_n ← adjacent_arcs_of(n)        {adjacent arcs of n}
  V_n ← parents_of(n)        {parent nodes of n in H_min}
  for all arc a ∈ A do
          {computation of possible adjacent arcs of n}
    t ← target_node_of(a)
    w⃗(t) ← w⃗(n) + w⃗(a)        {accumulated weight at t}
    if w⃗(t) ≤ L⃗ then
      if t ∉ V_n then
        A ← A ∪ {a}
      end if
    end if
  end for
end for
for all combination C(A) of the arcs in A do
  H ← H_min ∪ C(A)        {add the arcs of C(A) to H_min}
  S ← S ∪ {H}        {add H as new successor to S}
end for
```

The branch and bound algorithm is given as follows:

Algorithm 4 Branch and bound algorithm to find DMPSH with end to end constraints

Input: the valuated graph $G = (V, E)$, the weight vector $\overrightarrow{w}_e$ for each edge $e \in E$, the source node s, the destination set M and the end to end criterion $\overrightarrow{L}$
Output: H the DMPSH under the constraints if it exists

$\overrightarrow{w}(s) \leftarrow \overrightarrow{0}$ {initial value to accumulated weights}
$H_0 \leftarrow s$ {initialize the hierarchy with the source}
$d_0 \leftarrow 0$
$K \leftarrow \{(H_0, d_0)\}$ { K is the set of the leaves in the search tree}
while $K \neq \emptyset$ **do**
  select H_{min} the leaf in K with minimal length
  if $M \subset H_min$ **then**
    return H_{min} {it is the optimal solution}
    STOP
  end if
  $S \leftarrow$ *successor_hierarchies(H_{min})*
  compute S the set of successor hierarchies of H_{min}
  for all $H_s \in S$ **do**
    $d(H_s) \leftarrow$ number_of_hops(H_s)
    $K \leftarrow K \cup \{(H_s, d(H_s))\}$
  end for
  $K \leftarrow L \setminus (H_min, d(H_min))$ {delete H_{min}}
end while
STOP without solution

Let us notice that the problem can also be formulated as a Nonlinear Programming (NLP) problem. The case of a multi-channel, multi-objective multicast routing problem in MPLS network is modelled and discussed in [30]. A generalized framework can be found in [31] and a multi-objective evolutionary algorithm is proposed to solve it. The Nonlinear Programming formulation of the multi-objective optimization is a real alternative to model the problem but is out of scope of the recent paper.

4.4 Heuristic solutions

4.4.1 Some earlier proposed algorithms

The needs for multicast routing taking into account QoS parameters for multimedia applications have manifested early. A number of QoS routing algorithms based on single, dual and multiple metrics have been proposed. In [32] Steiner tree heuristics are modified to find trees limited on end to end delay and with less cost. The problem of finding minimal cost solution with constraint on the end to end delay was the aim of several investigations both for the NP hard unicast and multicast routing and several methods has been proposed [33] [34] [35] [36].

In [37] and [38] the authors study the problem of constructing a multicast tree that satisfies end to end delay constraint along the paths from the source to each destination with bounded variations among the delays along these paths. For this delay and delay variation-bounded Steiner tree problem the authors of [39] propose a heuristic multicast routing algorithm based on simulated annealing.

Following the classification found in [40], the QoS routing can be considered as a Restricted Shortest Path (or Shortest Tree) problem in the above cases. An other category of the QoS routing problem is the Bandwidth Restricted Path problem (only the unicast routing case is presented, but the same classification can also be applied on the multicast routing), when bandwidth is one of the constraints that must be satisfied by the path computation algorithm. Several references for theses problems are given in this survey work.

The most promising and the most expensive heuristic solution for multi-constrained multicast routing has been proposed in [24]. As it was resumed at the beginning of this section, the proposed heuristic, called Mamcra, constructs at first a shortest path from the source to each destination using the non linear length function $l(P)$ defined in 6. So, if there is a feasible solution for the multi-constrained multicast routing problem, then the constructed set of the shortest paths contains a feasible solution. Samcra algorithm realizing this first construction step is known as an algorithm which needs a non polynomial execution time. Let us notice here that the set of shortest paths which can contain cycles corresponds to a hierarchy. The topology of this hierarchy is a star. It is possible that it is different from the DMPSH. The projection of this directed hierarchy on the original graph may contain directed sub hierarchies which are also feasible under the given constraints. Trivially these sub hierarchies are more preferable because their length is less. The second step of Mamcra performs a greedy algorithm to obtain a multicast sub graph that uses as few links as possible. This second step eliminates a part of helpless redundancies using two properties of the obtained paths.

Property 1. Consider two feasible paths $P_1(s, d_1)$ and $P_2(s, d_2)$ from the source s to the destinations d_1 and d_2 respectively. If they form a cycle with the common node x and if $\overrightarrow{w}(P_2(s, d_2)) - \overrightarrow{w}(P_2(s, x)) + \overrightarrow{w}(P_1(s, x)) \leq^d \overrightarrow{L}$ then $P_2(s, d_2)$ may be replaced by the concatenation $P_1(s, x)P_2(x, d_2)$ without violating the constraints.

Property 2. Given a path $P(s, d)$ corresponding to the end to end constraints, let us suppose that it contains the node a. The sub path $P(s, a)$ also lies within the constraints but it is not necessary the shortest path from node s to node a. Consequently, in this case the shortest path may be replaced by $P(s, a)$. With other world: the shortest path may be eliminated from the path set.

Let us notice that Property 2 is a special case of Property 1. Using these properties, Mamcra aims the elimination of useless cycles from the set of shortest paths. It is clear that

1. the set of shortest paths does not always contain the DMPSH
2. even if the set of shortest paths contains the DMPSH, the greedy algorithm of Mamcra does not guarantee that all of useless paths will be deleted at the end of the procedure.

To obtain a more preferable directed multicast hierarchy, an improved cycle reduction algorithm (ICRA) and a tabu-search based meta-heuristic have been proposed in [28]. Similarly to Mamcra, these algorithms work also with the set of shortest paths and propose efficient cycle reductions..

Since the exact solution is intractable in some network topologies, a tradeoff between algorithm complexity (computation time) and performance of the resulted routes is needed. One of the possibilities resides in the relaxation of the tight constraints in trying to find good solution using linear cost metrics derived from the multiple weight values $w_i(x)$. As it is mentioned in [40], by combining a set of QoS metrics in a single (linear an additive) metric, it is possible to use existing polynomial time path computation algorithms such as Bellman-Ford or Dijkstra for heuristically solving the original QoS routing problem.

[41] proposes a good overview on the important QoS routing algorithms which try to found a path between a source and a destination node that satisfies a set of constraints. A study of work related to the cost/performance tradeoff of QoS routing is presented in [42]. Being an area of active research, QoS routing has a number of open issues that have been discussed as part of this later paper.

An original idea to solve the QoS routing problem resides in using a database containing existing routes and their QoS parameters. When a route request arrives, at first the database is interrogated and if a feasible route (eventually a feasible tree) is found, then it is proposed without hard route computation. So, different communications may be aggregated in the same tree. This kind of architecture, called Aggregated QoS Multicast (AQoSM), has been proposed by [43] to provide scalable and efficient QoS multicast in Diff-Serv networks. The combination of multi-constrained QoS multicast routing with tree aggregation has been proposed in [44].

4.4.2 Perspectives

In one hand, one can notice the problem of feasibility of the multicast routing. The here before analyzed optimization problem describe a real multicast routing problem with multiple tight QoS constraints. Today in real networks, the tight end to end conditions for multicast communication can not always be achieved for various reasons. At first, there are network states and conditions where feasible solution does not exist for a given route request. Even if the feasible solution exists, the computation time of the feasible solutions with tight criterion is expensive. In some real cases, the computation of the multicast routing structures (for instant, the proposed multicast "forest") can not be performed with a reasonable execution time tolerated by network operators. That is true, even if it is proved that often (in simple, ordinary topologies) the routing algorithm does not need important computation time [22]. Several propositions can be imagined to moderate these computational difficulties. Without exhaustive, let us enumerate some research issues in the following.

A first and simple possibility reside in applying more loose conditions for path selection. For example, by computing k shortest paths applying the simple hop distance and then by selecting the best feasible path (or the first path which is the closest to $\overrightarrow{L}$ if there is no feasible solution).

The complexity of the multiobjective optimal route research can be broken also by applying different (in preference linear) length functions to search shortest paths and trees.

One can use different dominance in the place of the tight Pareto's dominance. For example, the end to end QoS criterion can be sorted by their importance for the end user and lexicographical and other, more slight dominance can be applied [45].

Or, if the request is proved to be infeasible then some conditions may be relaxed (for example, more important delay or jitter can be authorized, or less bandwidth may be asked) knowing that this relaxation involve the degradation of the QoS (cf. this kind of proposition for example in [46]).

A very usefull idea can be the following. Supposing that the routing functionality of a designed controller unit (*e.g.* a Path Computation Element or PCE, cf. in [47]) can be polymorphic, different, polymorphic routing algorithms can be called to compute routes in function of constraints, topology, etc. For example a routing strategy may call the faster algorithms to compute simple multicast trees as SPT or approximated Steiner trees. If the given tree is within the QoS criterion, then it is used to route in the domain, else a second (polymorphic) algorithm is called to continue the route computation, etc. Originally, the idea of polymorphic routing was proposed in [48] for routing in MANETs.

5 Conclusions and perspectives

In this paper the optimal routing structure for source based multicast routing under different constraints was discussed. In graph theory, the question corresponds to finding minimal partial spanning structures for a given set of nodes. We pointed here that the optimal structure is a hierarchy. Generally, this hierarchy does not correspond to a tree. Spanning trees are special cases of spanning hierarchies. We analyzed two specific optimization problems.

In our first study, the optimal multicast routing problem was formulated in optical networks using WDM technique. For simplicity reason, the case of WDM networks without wavelength converters and with sparse splitting capable switches was analyzed. We also supposed that the objective is to minimize the overall cost of the partial spanning structure. We showed that the optimal structure is a directed hierarchy and its computation is NP-complete. Two intractable, exact algorithms and a heuristic solution were proposed. In future optical networks, multicast routing should be performed in various cases with different conditions. Future works should analyze the optimal hierarchies and the design of efficient heuristics in the case of sparse wavelength converters and splitters. An interesting challenge consists to add constraints on the diameter of routing structures since the propagation of the light without amplification is limited.

In optical networks without wavelength converters and splitters, the multicast should be performed using only lightpaths and permitting the O/E/O conversions in some nodes. Since the O/E/O conversion and the process in electrical domain add important latencies, hierarchies with few number of branching nodes are needed in this kind of networks.

Our second study told a different case of constraints, where the QoS related constraints are given between the source node and any destinations. We analyzed the optimal routing problem with multiple QoS constraints given by additive metrics. We demonstrated that the optimal partial spanning structure in this case is also a hierarchy and the computation of the optimal directed hierarchy is intractable. An exact branch and bound algorithm has been presented to compute the optimal solution. In real routing cases, the computation of the optimal structures can not be tolerated by network operators. Several propositions can be imagined to moderate these difficulties. We enumerated some research issues for future works. A first and simple possibility reside in applying more loose conditions for path selection. For example, by computing k shortest paths applying the simple hop distance and then by selecting the best feasible path or first path which is the closest to $\overrightarrow{L}$ if there is no feasible solution. The complexity of the multiobjective optimal route research can be broken also by applying different (in preference linear) length functions to search shortest paths and trees, One can use different dominance in the place of the tight Pareto's dominance. For example, the end to end QoS criterion can be sorted by their importance for the end user and lexicographical or other, more slight dominance can be applied.

The optimization of multicast routing structures can arise in different contexts and for different reasons. An interesting optimization problem of multicast routing under constraints is given when computing the optimal routing structure for explicit tree-based multicast routing. In this case, the routing structure is encoded in the header of the transmitted messages. So, the size of the routing structure (the number of the significant nodes) has a direct impact on the communication cost. Furthermore the packet size being limited, the routing structure corresponds to a hierarchy in which some edges are used several times.

The optimization problems of multicasting under different constraints correspond to interesting challenges.

References

[1] Bollobás, B.: Modern Graph Theory. Springer-Verlag, New York, NY, USA (1998)

[2] Distel, R.: Graph Theory, Third Edition. Springer-Verlag, Heidelberg (2005)

[3] David, R., Alla, H.: Discrete, Continuous, and Hybrid Petri Nets. Springer-Verlag, New York, NY, USA (2005)

[4] Hwang, F.K., Richards, D.S., Winter, P.: The Steiner Tree Problem. Annals of Discrete Mathematics **53** (1992)

[5] Xue, G.R., Yang, Q., Zeng, H.J., Yu, Y., Chen, Z.: Exploiting the hierarchical structure for link analysis. In: SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM (2005) 186–193

[6] Médard, M., Finn, S.G., Barry, R.A., Gallager, R.G.: Redundant Trees for Preplanned Recovery in Arbitrary Vertex-Redundant or Edge-Redundant Graphs. IEEE/ACM Transactions on Networking **7** (1999) 641–652

[7] Murthy, C.S.R., Gurusamy, M.: WDM Optical Networks: Concepts, Design and Algorithms. Prentice Hall (2001)

[8] Zhang, X., Wei, J., Qiao, C.: Constrained multicast routing in WDM networks with sparse light splitting. In: IEEE INFOCOM. Volume 3. (2000) 1781–1790

[9] Hakimi, S.: Steiner problem in graphs and its implications. Networks **1** (1971) 113?–133

[10] Lawler, E.: Combinatorial Optimization. Networks and Matroids. Holt, Rinehart and Winston, New York, NY, USA (1976)

[11] Balakrishnan, A., Patel, N.R.: Problem reduction methods and a tree generation algorithm for the Steiner network problem. Netw. **17** (1987) 65–85

[12] Singhal, N.K., Sahasrabuddhe, L.H., Mukherjee, B.: Optimal multicasting of multiple light-trees of different bandwidth granularities in a WDM mesh network with sparse splitting capabilities. IEEE/ACM Trans. Netw. **14** (2006) 1104–1117

[13] Ramanathan, S.: Multicast tree generation in networks with asymmetric links. IEEE/ACM Trans. Netw. **4** (1996) 558–568

[14] Libeskind-Hadas, R., Melhem, R.: Multicast routing and wavelength assignment in multihop optical networks. IEEE/ACM Trans. Netw. **10** (2002) 621–629

[15] Hartline, J.R.K., Libeskind-Hadas, R., Dresner, K.M., Drucker, E.W., Ray, K.J.: Optimal virtual topologies for one-to-many communication in WDM paths and rings. IEEE/ACM Trans. Netw. **12** (2004) 375–383

[16] Miller, E., Libeskind-Hadas, R., Barnard, D., Chang, W., Dresner, K., Turner, W., Hartline, J.: On the complexity of virtual topology design for multicasting in WDM trees with tap-and-continue and multicast-capable switches. Selected Areas in Communications, IEEE Journal on **22** (Nov. 2004) 1601–1612

[17] Ferrel, I., Mettler, A., Miller, E., Libeskind-Hadas, R.: Virtual topologies for multicasting with multiple originators in WDM networks. IEEE/ACM Trans. Netw. **14** (2006) 183–190

[18] Manjunath, P.M.: (Effect of objective function on virtual topology design in optical networks)

[19] Zsigri, A., Guitton, A., Molnár, M.: Construction of Light-trees for WDM Multicasting under Splitting Capability Constraints. In: 10th International Conference on Telecommunications. (2003)

[20] Zsigri, A., Guitton, A., Molnár, M.: Two Multicast Algorithms for Sparse Splitting Capable Networks. In: 7th IFIP Working Conference on Optical Network Design and Modelling. (2003)

[21] Takahashi, H., Matsuyama, A.: An approximate solution for the Steiner problem in graphs. Mathematica Japonica **24** (1980) 573–577

[22] Mieghem, P.V., Kuipers, F.A.: Concepts of exact QoS routing algorithms. IEEE/ACM Trans. Netw. **12** (2004) 851–864

[23] Wang, B., Hou, J.C.: Multicast Routing and its QoS Extension: Problems, Algorithms and Protocols. IEEE Network **14** (2000) 22–36

[24] Kuipers, F.A., Mieghem, P.V.: MAMCRA: a constrained-based multicast routing algorithm. Computer Communications **25** (2002) 802–811

[25] Mieghem, P.V., Neve, H.D., Kuipers, F.: Hop-by-hop quality of service routing. Comput. Networks **37** (2001) 407–423

[26] Davie, B.S., Rekhter, Y.: MPLS: Technology and Applications. Morgan Kaufmann (2000)

[27] Boivie, R., Feldman, N., Metz, C.: Small group multicast: a new solution for multicasting on the internet. IEEE Internet Computing **4** (2003) 75 – 79

[28] Ali, N.B., Molnár, M., Belghith, A.: Multi-constrained QoS Multicast Routing Optimization. Technical report, IRISA, Rennes, France (2008)

[29] Wang, Z., Crowcroft, J.: Quality-of-Service Routing for Supporting Multimedia Applications. IEEE Journal of Selected Areas in Communications **14** (1996) 1228–1234

[30] Meisel, Y.D., Fabregat, R., Fàbrega, L.: Multi-objective scheme over multi-tree routing in multicast MPLS networks. In: LANC '03: Proceedings of the 2003 IFIP/ACM Latin America conference on Towards a Latin American agenda for network research, New York, NY, USA, ACM (2003) 59–71

[31] Fabregat, R., Donoso, Y., Baran, B., Solano, F., Marzo, J.L.: Multi-objective optimization scheme for multicast flows: a survey, a model and a MOEA solution. In: LANC '05: Proceedings of the 3rd international IFIP/ACM Latin American conference on Networking, New York, NY, USA, ACM (2005) 73–86

[32] Kompella, V.P., Pasquale, J.C., Polyzos, G.C.: Multicast routing for multimedia communication. IEEE/ACM Trans. Netw. **1** (1993) 286–292

[33] Salama, H.F., Reeves, D.S., Viniotis, Y.: Evaluation of Multicast Routing Algorithms for Real-Time Communication on High-Speed Networks. IEEE Journal on Selected Areas in Communications **15** (1997) 332–345

[34] Hać, A., Zhou, K.: A new heuristic algorithm for finding minimum-cost multicast trees with bounded path delay. Int. J. Netw. Manag. **9** (1999) 265–278

[35] Costa, L.H.M.K., Fdida, S., Duarte, O.: Hop by hop multicast routing protocol. In: SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, New York, NY, USA, ACM (2001) 249–259

[36] Guo, L., Matta, I.: Search space reduction in QoS routing. Comput. Networks **41** (2003) 73–88

[37] Prasad, T., Hansdah, R.C.: An efficient distributed multicast routing protocol with delay and delay variation constraints. In: ICCC '02: Proceedings of the 15th international conference on Computer communication, Washington, DC, USA, International Council for Computer Communication (2002) 955–971

[38] Rouskas, G.N., Baldine, I.: Multicast routing with end-to-end delay and delay variation constraints. Technical report, North Carolina State University at Raleigh, Raleigh, NC, USA (1995)

[39] Kun, Z., Yong, Q., Hong, Z.: Dynamic multicast routing algorithm for delay and delay variation-bounded Steiner tree problem. Know.-Based Syst. **19** (2006) 554–564

[40] Masip-Bruin, X., Yannuzzi, M., Domingo-Pascual, J., Fonte, A., Curado, M., Monteiro, E., Kuipers, F., Mieghem, P.V., Avallone, S., Ventre, G., Aranda-Gutierrez, P., Hollick, M., Steinmetz, R., Iannone, L., Salamatian, K.: Research Challenges in QoS Routing. Computer Communications, NoE E-Next special issue **29** (2006) 563–581

[41] Mieghem, P.V., Kuipers, F., Korkmaz, T., Krunz, M., Monteiro, M.C.E., Masip-Bruin, X., Solé-Pareta, J., Sánchez-López, S.: 3. In: Quality of Future Internet Services (COST Action 263 Final Report). Volume 2856 of Lecture Notes in Computer Science. Springer, Berlin (2003) 80–117

[42] Paul, P., Raghavan, S.V.: Survey of QoS routing. In: ICCC '02: Proceedings of the 15th international conference on Computer communication, Washington, DC, USA, International Council for Computer Communication (2002) 50–75

[43] Cui, J.H., Lao, L., Faloutsos, M., Gerla, M.: AQoSM: scalable qos multicast provisioning in Diff-Serv networks. Comput. Networks **50** (2006) 80–105

[44] Ben Ali, N., Moulierac, J., Belghith, A., Molnár, M.: mQMA: Multi-constrained QoS Multicast Aggregation. In: IEEE Globecom. (2007)

[45] Miettinen, K.: Nonlinear Multiobjective Optimization. Kluwer Academic Publishers (1999)

[46] Liebeherr, J., Liao, D.: A service with bounded degradation in quality-of-service networks. In: INFOCOM '95: Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communication Societies (Vol. 3)-Volume, Washington, DC, USA, IEEE Computer Society (1995) 1103

[47] Farrel, A., Vasseur, J.P., Ash, J.: A Path Computation Element (PCE)-Based Architecture, Request for Comments: RFC4655 (2007)

[48] Ben Mnaouer, A., Chen, L., Foh, C.H., Tantra, J.W.: OPHMR: An Optimized Polymorphic Hybrid Multicast Routing Protocol for MANET. IEEE Transactions on Mobile Computing **6** (2007) 551–562

Contents